

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
Факультет інформатики та обчислювальної техніки
Кафедра автоматики та управління в технічних системах**

«До захисту допущено»

Завідувач кафедри

_____ О.І. Ролік

«__» _____ 2019 р.

**Дипломний проект
на здобуття ступеня бакалавра
з напрямку підготовки 6.050103 «Програмна інженерія»
на тему: «Браузерна онлайн гра на архітектурі клієнт-сервер»**

Виконав (-ла):

студент (-ка) IV курсу, групи ІТ-51

Корзунов Дмитро Андрійович _____

Керівник:

Асистент Міщенко Володимир Олександрович _____

Рецензент:

д.т.н., професор, Пархомей Ігор Ростиславович _____

Засвідчую, що у цьому дипломному
проекті немає запозичень з праць інших
авторів без відповідних посилань.

Студент (-ка) _____

Київ – 2019 рік

**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра автоматики та управління в технічних системах**

Рівень вищої освіти – перший (бакалаврський)

Напрямок підготовки – 6.050103 «Програмна інженерія»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ О.І. Ролік

«___» _____ 2019 р.

ЗАВДАННЯ

на дипломний проект студенту

Корзунову Дмитру Андрійовичу

1. Тема проекту «Браузерна онлайн гра на архітектурі клієнт-сервер», керівник проекту асистент Міщенко Володимир Олександрович, затверджені наказом по університету від «___» _____ 2019 р. № _____

2. Термін подання студентом проекту _____

3. Вихідні дані до проекту ___ JavaScript, Node.js, HTML 5, PostgreSQL, Visual Studio Code, Mocha.JS

4. Зміст пояснювальної записки ___ Вступ. 1. Огляд існуючих архітектур. 2. Огляд існуючих програмних рішень та технологій. 3. Програмна реалізація додатку. 4. Інструкція користувача браузерної онлайн гри на архітектурі клієнт-сервер. Висновки

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслеників, плакатів, презентацій тощо) _____ 1. Діаграма послідовності – плакат. 2. Діаграма використання – плакат. 3. Діаграма діяльності – плакат. 4. Діаграма бази даних – плакат.

7. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів проекту	Примітка
1	Отримання завдання	25.02.2019	
2	Збір інформації	05.03.2019	
3	Вивчення варіантів реалізації та вибір варіанту для розробки	10.03.2019	
4	Модифікація бази даних для гри	19.03.2019	
5	Розробка серверної частини	28.03.2019	
6	Розробка інтерфейсу користувача	18.04.2019	
7	Тестування роботи	27.04.2019	
8	Оформлення дипломної роботи	31.05.2019	
9	Отримання допуску до захисту	04.06.2019	
10	Захист дипломної роботи	18.06.2019	

Студент

Д.А. Корзунов

Керівник проекту

В.О. Міщенко

АНОТАЦІЯ

Корзунов Д.А. Браузерна онлайн гра на архітектурі клієнт-сервер. КПП ім. Ігоря Сікорського, Київ, 2019.

Проект містить 65 сторінок тексту, 34 рисунків, 3 таблиці, посилання та 21 літературних джерел, додатки.

Ключові слова: браузерна гра, клієнт-сервер, архітектура, MMORPG, веб-інтерфейс.

Об'єктом дослідження є процес організації розвитку людей в повсякденному житті.

Мета роботи – є підвищення достовірності та ефективності розробки браузерної онлайн-гри шляхом використання архітектури клієнт-сервер.

У дипломному проекті вирішується задача організації ігрового процесу та розвитку людей за допомогою браузерної онлайн гри. Для реалізації програмного засобу, обрані сучасні інформаційні технології. Порівняно існуючі архітектури та вибрано найбільш підходящу, спроектовано розподілену базу даних для зберігання даних гри та користувачів, проведено тестування реалізованого програмного засобу.

Результати дипломної роботи можуть бути застосовані для покращення організації відпочинку та розвитку у вищих навчальних закладах.

SUMMARY

Korzunov DA Browser-based online game on client-server architecture. KPI them Igor Sikorsky, Kyiv, 2019.

The project contains 65 pages of text, 34 figures, 3 tables, links and 21 literary sources, applications.

Keywords: browser game, client server, architecture, MMORPG, web interface.

The object of research is the process of organizing the development of people in everyday life.

The purpose of the work is to increase the reliability and efficiency of developing browser-based online games by using client-server architecture.

The graduation project solves the problem of organizing the game process and human development through browser-based online games. For the implementation of the software, selected modern information technology. Comparable existing architectures and the most suitable, designed distributed database for storing game data and users, the testing of the implemented software has been selected. The results of the graduation project may be used to improve the scientific activity and educational process in higher education institutions.

The results of the thesis may be used to improve the organization of recreation and development in higher education institutions.

Ном. поз.	Формат	Позначення	Найменування	Кількість аркушів	№. екз.	Примітки
1			Документація загальна			
2						
3			Знову розроблена			
4						
5						
6						
7	A4	IT51.060БАК.002 ПЗ	Пояснювальна записка	65		
8						
9	A3	IT51.060БАК.003 Д1	Діаграма послідовності	1		
10						
11	A3	IT51.060БАК.003 Д2	Діаграма використання	1		
12						
13	A3	IT51.060БАК.003 Д3	Діаграма діяльності	1		
14						
15	A3	IT51.060БАК.003 Д4	ER – діаграма	1		
16						
17						
18						
19						
20						
21						
22						
23						
24						
25						
26						

					IT51.060БАК.001 ТП		
Змн.	Арк.	Шуквич М.К.	Підпис	Дат			
Розроб.		Корзунов Д.А.			Браузерна онлайн гра на архітектурі клієнт-сервер		
Перевір.		Міщенко В.О.					
Реценз.							
Н. Контр.							
Затверд.							
					Літ.	Арк.	Аркушів
					НТУУ "КПІ" ФІОТ Група ІТ-51		

Пояснювальна записка
до дипломного проекту
на тему: «Браузерна онлайн гра на архітектурі
клієнт-сервер»

Київ – 2019 рік

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ	5
ВСТУП	6
1 ОГЛЯД ІСНУЮЧИХ АРХІТЕКТУР.....	9
1.1 Інформаційна технологія.....	9
1.2 Визначення.....	10
1.3 Опис моделі клієнт-сервер	12
1.4 Переваги та недоліки архітектури клієнт-сервер	14
1.5 Типи клієнт-серверних архітектур	16
1.6 Двокаскадна архітектура	17
1.7 Трьохкаскадна архітектура	19
1.8 Багатокаскадна архітектура	20
1.9 Порівняння архітектур.....	21
1.10 Клієнт-серверні технології.....	22
Висновки до розділу	23
2. ОГЛЯД ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ ТА ТЕХНОЛОГІЙ.....	24
2.1 Браузерна онлайн-гра загальні відомості	24
2.2 Браузерні онлайн-ігри для одного користувача.....	25
2.3 Браузерні онлайн-ігри розраховані на велику кількість користувачів....	25
2.4 Браузерні онлайн-ігри масово-багатокористувацькі.....	26
2.5 Порівняння жанрів браузерних онлайн-ігр	27
2.6 Огляд браузерних онлайн-ігор жанру ММОРПГ	29

					IT51.090БАК.002 ПЗ		
Змн.	Арк.	№ докум.	Підпис	Дат	Браузерна онлайн гра на архітектурі клієнт- сервер		
Розроб.		Корзунов Д.А.					
Перевір.		Міщенко В.О.					
Реценз.							
Н. Контр.							
Затверд.							
					Літ.	Арк.	Аркушів
						2	69
					НТУУ “КПІ” ФІОТ Група IT-51		

2.7 Технологія WebAssembly	31
2.8 Технологія HTML5	33
2.9 Технологія Unity3D.....	34
2.10 Компоненти браузерної онлайн-гри.....	36
Висновки до розділу	37
3. ПРОГРАМНА РЕАЛІЗАЦІЯ ДОДАТКУ	39
3.1 HTML5 і Node.js: багатокористувацька браузерна онлайн-гра на архітектурі клієнт-сервер	39
3.2 Процес браузерної онлайн-гра на архітектурі клієнт-сервер	40
3.3 Діаграма варіантів використання для програмного забезпечення браузерної онлайн-гра на архітектурі клієнт-сервер.....	42
3.4 Діаграма класів для програмного забезпечення браузерної онлайн-гра на архітектурі клієнт-сервер	Ошибка! Закладка не определена.
3.5 Діаграма послідовності для програмного забезпечення браузерної онлайн-гра на архітектурі клієнт-сервер	43
3.6 Діаграма діяльності для програмного забезпечення браузерної онлайн-гра на архітектурі клієнт-сервер	45
3.7 Діаграма розгортання для програмного забезпечення браузерної онлайн-гра на архітектурі клієнт-сервер.....	47
3.8 Статичний аналіз коду програмного забезпечення браузерної онлайн-гри на архітектурі клієнт-сервер	47
Висновок	49
4 ІНСТРУКЦІЯ КОРИСТУВАЧА БРАУЗЕРНОЇ ОНЛАЙН ГРИ НА ОСНОВІ АРХІТЕКТУРИ КЛІЄНТ-СЕРВЕР.....	51
4.1 Призначення браузерної онлайн-гри на основі архітектури клієнт-сервер	51

4.2 Завантаження браузерної онлайн-гри на основі архітектури клієнт-сервер	51
4.3 Початковий інтерфейс браузерної онлайн-гри на основі архітектури клієнт-сервер.....	52
4.4 Інтерфейс локації браузерної онлайн-гри на основі архітектури клієнт-сервер.....	53
4.5 Інтерактивні елементи браузерної онлайн-гри на основі архітектури клієнт-сервер.....	54
4.6 «Логи» браузерної онлайн-гри на основі архітектури клієнт-сервер.....	55
4.7 Інвентар в браузерної онлайн-гри на основі архітектури клієнт-сервер.	57
4.8 Чат користувачів в браузерної онлайн-гри на основі архітектури клієнт-сервер.....	58
Висновок	59
ВИСНОВКИ.....	61
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	63
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ....	Ошибка! Закладка не определена.

					<i>IT51.090БАК.002 ПЗ</i>	Аркуш
Зм.	Арк.	№ документа	Підпис	Дат		4

ПЕРЕЛІК СКОРОЧЕНЬ

MVC	–	Model View Controller
БД	–	база даних
ІТ	–	інформаційна технологія
КС	–	комп'ютерна система
КМ	–	комп'ютерна мережа
ОС	–	операційна система
ПЗ	–	програмне забезпечення
REST	–	REpresentational State Transfer
SOAP	–	Simple Object Access Protocool
JSON	–	JavaScript Object Notation
XML	–	eXtensible Markup Language
HTTP	–	Hyper Text Transfer Protocol
RMI	–	Remote Method Invocation
CORBA	–	Common Object Request Broker Architecture
EE	–	Enterprise Edition
JPA	–	Java Persistence API
EJB	–	Enterprise Java Beans
ПП	–	програмний продукт

ВСТУП

Технології швидко змінюють наш світ і стають невід'ємною частиною нашого повсякденного життя. Ми вступаємо в найцікавішу епоху інформатики і техніки в історії людства. Інформація та технології розвиваються і перетворюються на складні системи. Багато кордонів між наукою, технікою та соціальними системами перехресно пов'язані з поєднанням знань, інструментів і додатків. Це час, коли ми з великою точністю розглядаємо, як все працює на системному рівні з минулого, теперішнє у майбутньому. Сьогодні інформаційні технології спрямовані на розуміння проблем з точки зору зацікавлених сторін і вивчення технічних основ, соціальних та теоретичних наслідків, онлайн-баз даних, широкого використання баз даних в уряді, промисловості та освіті, а також розвиток Інтернету та світу Wide Web. Це час, коли фізичні, біологічні та соціальні науки об'єднують сили з інформаційними науками та технологіями. Це час, коли ми зробили та зробимо надзвичайний прогрес в історії людства через технології та інформаційні науки.

Архітектурний стиль, іноді званий архітектурним шаблоном - це набір принципів, високорівнева схема, що забезпечує абстрактну інфраструктуру для сімейства систем.

Архітектурний стиль покращує секціонування і сприяє повторному використанню дизайну завдяки забезпеченню рішень проблем, що часто зустрічаються. Архітектурні стилі і шаблони можна розглядати як набір принципів, які формують додаток.

«Клієнт/серверна» архітектура – це система, що поділяється на два додатки, де клієнт виконує запити до сервера. У багатьох випадках в ролі сервера виступає база даних, а логіка додатка представлена процедурами зберігання.

Актуальність роботи полягає в розробці браузерної онлайн-гри на основі архітектури клієнт-сервера. Розробка браузерної онлайн-гри має здійснюватися

					<i>IT51.090БАК.002 ПЗ</i>	Аркуш
						6
Зм.	Арк.	№ документа	Підпис	Дат		

без скачування гігабайтних файлів і не потребує встановлювання ігрових процесів, які відбуваються в браузері комп'ютера.

Метою дипломної роботи є підвищення достовірності та ефективності розробки браузерної онлайн-гри шляхом використання архітектури клієнт-сервер.

Поставлена мета досягається розв'язанням таких основних задач:

- необхідно розробити метод розробки браузерної онлайн-гри шляхом використання архітектури клієнт-сервер;

- метод повинен використовувати архітектуру клієнт-сервер для зниження ресурсоемності процесу завантаження під час онлайн-гри;

- метод повинен виявляти нові ШПЗ типу backdoor на основі обробки зібраних в мережі та в комп'ютерній системі множини поведінок ШПЗ типу backdoor, а класифікацію поведінок здійснювати шляхом виділення підмножини зібраних ознак і генерації таких необхідних поведінок, які дозволять виявити ШПЗ типу backdoor;

Наукова новизна отриманих результатів:

- набуття подальшого використання браузерної онлайн-гри шляхом використання архітектури клієнт-сервер;

- набуття подальшого розвитку інформаційної технології браузерної онлайн-гри шляхом використання архітектури клієнт-сервер.

Метод використовує архітектуру клієнт-сервер, що дозволяє здійснювати процес браузерної онлайн-гри без скачування гігабайтних файлів і не потребує встановлювання ігрових процесів, які відбуваються в браузері комп'ютера.

Інформаційна технологія дозволяє розробити браузерну онлайн-гру на основі архітектури клієнт-сервер.

Практична цінність отриманих результатів. В результаті виконаного наукового дослідження розроблена архітектура і компоненти програмного забезпечення, що реалізує браузерну онлайн-гру на основі архітектури клієнт-сервер.

					IT51.090БАК.002 ПЗ	Аркуш
						7
Зм.	Арк.	№ документа	Підпис	Дат		

У даній роботі викладено вимоги до методології розробки браузерної онлайн-гри на основі архітектури клієнт-сервер.

Для розв'язання поставлених задач по розробці браузерної онлайн-гри використовуються основні положення системного аналізу, методу обчислювального інтелекту, системне програмування, методів аналізу даних, теорії дискретної математики та теорії комп'ютерних мереж.

					IT51.090БАК.002 ПЗ	Аркуш
						8
Зм.	Арк.	№ документа	Підпис	Дат		

1 ОГЛЯД ІСНУЮЧИХ АРХІТЕКТУР

Першим етапом розробки програмного забезпечення є аналіз та огляд існуючих архітектур. Розгляд практичності та актуальності застосування тої чи іншої архітектури.

Також для більшого розуміння було зроблено декілька UML діаграм, які відображають можливості та функціонал системи. За допомогою цього можемо проаналізувати предметну область та визначити, що саме повинна виконувати розроблювана система

1.1 Інформаційна технологія

Інформаційна технологія - це процес, що використовує всі засоби та методи збору, обробки та передачі даних для отримання інформації про стан об'єкта, процесу або явища. Ціль інформаційної технології - виробництво інформації.

Удосконалення комп'ютерних мереж в інформаційній сфері та застосування телекомунікаційних засобів зв'язку визначено новим етапом розвитку інформаційної технології. Нова інформаційна технологія(ІТ) - це ІТ з «дружелюбним» інтерфейсом роботи користувача, що використовує персональні комп'ютери(ПК) та телекомунікаційні засоби. Нова інформаційна технологія базується на наступних основних принципах.

- Інтегрованість з іншими програмними продуктами.
- Гнучкість процесу зміни даних і постановок завдань.
- Інтерактивний (діалоговий) режим роботи з комп'ютером.

В якості інструментів інформаційної технології використовуються поширені види програмних продуктів: текстові процесори, електронні таблиці, системи управління базами даних, електронні календарі, інформаційні системи.

До основних видів інформаційних технологій стосується наступних.

					ІТ51.090БАК.002 ПЗ	Аркуш
						9
Зм.	Арк.	№ документа	Підпис	Дат		

– Інформаційна технологія обробки даних призначена для вирішення добре структурованих завдань, алгоритми рішень яких добре відомі і для вирішення яких є всі необхідні вихідні дані. Ця технологія застосовується на рівні виконавчої діяльності персоналу невисокої кваліфікації в цілях автоматизації деяких рутинних, постійно повторюваних операцій управління.

– Інформаційна технологія управління призначена для інформаційного обслуговування всіх працівників підприємств, пов'язаних з прийняттям управлінських рішень. Дана інформація зазвичай представляється у вигляді регулярних або спеціальних управлінських звітів і містить дані про минулий, майбутній і можливе майбутнє підприємства.

– Інформаційна технологія автоматизованого офісу привертає до уваги існуючу систему зв'язку персоналу підприємства. Автоматизація офісу передбачає організацію та підтримку комунікаційних процесів, як у фірмі, так і з зовнішніми обсягами комп'ютерних мереж та інших сучасних засобів передачі і роботи з інформацією.

– Інформаційна технологія підтримки прийняття рішень призначена для вироблення управлінських рішень, що відбуваються в результаті ітераційного процесу, в якому учасники допомагають прийняти рішення (виводячи на облік і оцінювати отриманий результат).

– Інформаційна технологія експертних систем заснована на використанні мистецького інтелекту. Експертні системи дають можливість менеджера отримувати консультації експертів по будь-якій проблемі, про які в цих системах накопичуються знання.

1.2 Визначення

З цієї метою постало питання вибору архітектури програмного забезпечення, що реалізуватиме браузерну онлайн-гру.

					IT51.090БАК.002 ПЗ	Аркуш
						10
Зм.	Арк.	№ документа	Підпис	Дат		

Для реалізації інформаційної технології розроблення браузерної онлайн-гри було побудовано ПЗ.

З цією метою було розроблено архітектуру програмного забезпечення, яка являє собою сукупність найважливіших рішень про організацію програмної системи. Вона містить в собі:

- структурні елементи і їх інтерфейси;
- з'єднання елементів у все більш крупні системи;
- архітектурний стиль, який визначає спосіб організації елементів, і їх з'єднань.

Архітектура інформаційної технології ПЗ, як зазначалося раніше, є одним з важливих об'єктів проектування програмних систем. Архітектурний вигляд складається з двох компонентів:

- елементи;
- відносини між елементами.
- Архітектурні види можна розділити на три основні типи:
 - модульні (module views), які представляють систему як структуру з різних програмних блоків;
 - компоненти-і-коннектори (component-and-connector views) - показують систему як структуру з паралельно запущених елементів (компонентів) і способів їх взаємодії (конекторів);
 - розміщення (allocation views), що відображає розміщення елементів системи в зовнішніх середовищах.

Архітектурний стиль, іноді званий архітектурним шаблоном - це набір принципів, високорівнева схема, що забезпечує абстрактну інфраструктуру для сімейства систем.

Архітектурний стиль покращує секціонування і сприяє повторному використанню дизайну завдяки забезпеченню рішень проблем, що часто зустрічаються. Архітектурні стилі і шаблони можна розглядати як набір

					IT51.090БАК.002 ПЗ	Аркуш
						11
Зм.	Арк.	№ документа	Підпис	Дат		

принципів, які формують додаток.

Архітектура інформаційної комп'ютерної - будується на основі апаратної частини (ЕОМ), телекомунікаційного і програмного забезпечення.

Інформаційне забезпечення - це сукупність методів і засобів щодо розміщення й організації інформації, що містять системи класифікації і кодування, уніфіковані системи документації, раціоналізації документообігу і форм документів, методів створення внутрішньомашинної інформаційної бази ІС.

Масив - зображає набір даних одної форми з усіма їх значеннями або групи таких наборів даних, що належать до однієї задачі підсистеми.

Технічне забезпечення - це комплекс технічних засобів, що застосовується для функціонування системи обробки даних.

Архітектура «клієнт-сервер» - визначає загальні принципи організації взаємодії в мережах, де знаходяться сервери, вузли-постачальники певних специфічних функцій (сервісів) і клієнтів, які користувалися цими функціями.

Сервер – це програма, що надає деякі послуги іншим програмам і обслуговує запити клієнтів на отримання ресурсів певного виду.

Клієнт – це програма, що використовує послугу, надану програмою сервера.

1.3 Опис моделі клієнт-сервер

Архітектури “клієнт-сервер” широко використовуються в сучасних програмних продуктах. Їхнє використання забезпечує більш високу швидкість виконання, можливість розділити обчислювальні навантаження на різні машини, а також можливість розробити кілька інтерфейсів користувача, що відносяться до комп'ютерних систем.

«Клієнт/серверна» архітектура – це система, що поділяється на два додатки, де клієнт виконує запити до сервера (рисунок - 1.1). У багатьох випадках в ролі сервера виступає база даних, а логіка додатка представлена процедурами зберігання.

					IT51.090БАК.002 ПЗ	Аркуш
						12
Зм.	Арк.	№ документа	Підпис	Дат		

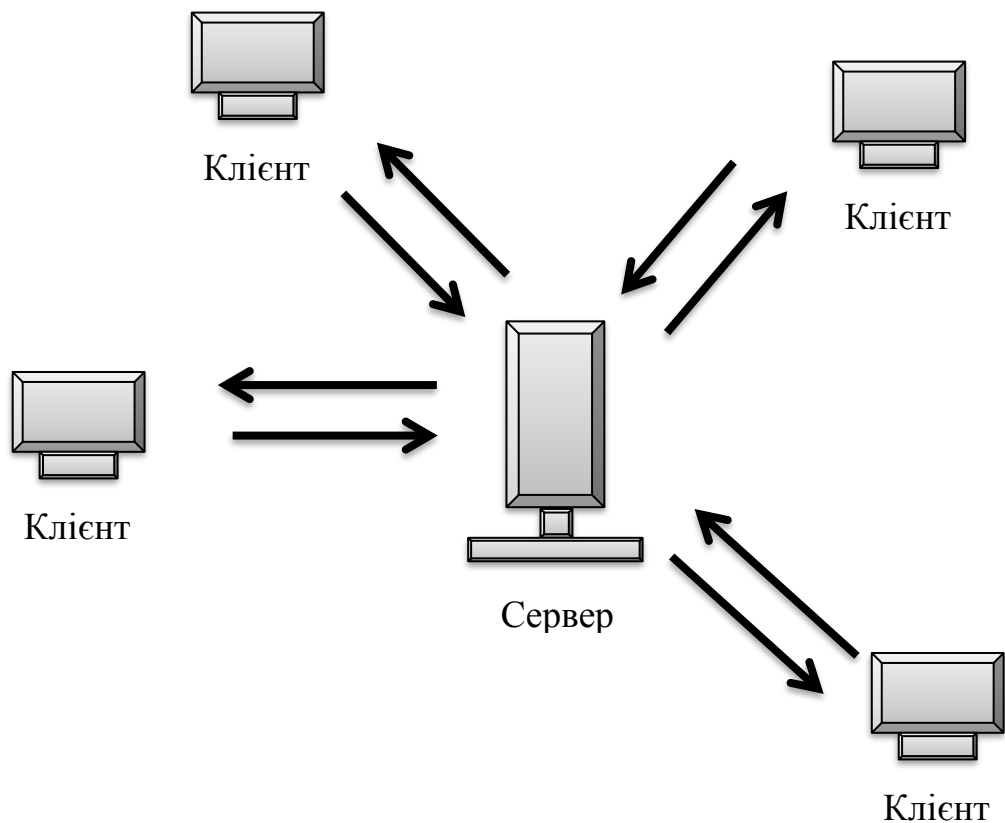


Рисунок 1.1 – Архітектура «клієнт-сервер»

Архітектура «клієнт-сервер» визначає загальні принципи організації взаємодії в мережі, де є сервери, вузли-постачальники деяких специфічних функцій (сервісів) і клієнти, споживачі цих функцій (рисунок - 1.2).

У даній роботі досліджено розвиток середовища, в якому користувач зможе легко реалізувати розподілені алгоритми для різних аплікацій. Таке середовище повинно забезпечувати відповідні засоби для зв'язку між процесами, що використовуються на різних комп'ютерах, так, щоб користувач міг зосередити первинні питання щодо алгоритму розробки браузерної онлайн-гри. Іншими словами, необхідно забезпечити прозорість щодо мережі зв'язку. Система може бути використана для розробки і випробування алгоритмів розподілу, особливо алгоритмів, що стосуються комп'ютерних мереж.

Можливості, що надаються користувачеві для вирішення розподілених задач, включають:

- Ряд примітивних викликів, що використовуються для зв'язку між процесами, що складають розподілений алгоритм.
- Інструменти для переміщення та виконання процесів на потрібних машинах в простій і ефективній формі.
- Засоби вимірювання часу виконання алгоритмів, що дозволяють зробити правильні висновки, або теоретичні результати можуть бути забезпечені.

Всі вищезазначені пункти повинні бути запропоновані інтегрованим і вільним чином. Концепція інтеграції задумана таким чином, що всі необхідні засоби розробки (наприклад, редактори, компілятори, ігри тощо) надаються одночасно і підлягають вибору користувача. Концепція “дружелюбності” відноситься до інтерфейсу між користувачем і засобом розробки.



Рисунок 1.2 – Архітектурний шаблон програмного забезпечення «клієнт-сервер»

1.4 Переваги та недоліки архітектури клієнт-сервер

З одного боку, архітектура «клієнт-сервер» являє собою природне рішення з точки зору виробника, так як в ній використовуються всі доступніші мікрокомп'ютери і мережі. З іншого боку, архітектура «клієнт-сервер», можливо, є ідеальним вибором для підтримки обраного організацією напрямки бізнесу.

Практичні реалізації такої архітектури називаються «клієнт-серверними» технологіями. Кожна технологія визначає власні або використовує наявні правила взаємодії між клієнтом і сервером, які називаються протоколом обміну (протоколом взаємодії).

У таблиці 1.1 показано, що користувачі при переході на архітектуру ПЗ «клієнт-сервер», крім одержуваних переваг, повідомляються про численні проблеми.

Проте завдяки зниженню вартості і зростання популярності персональних комп'ютерів, а також завдяки зростаючій конкуренції в промисловості, архітектура «клієнт-сервер» в доступному для огляду майбутньому буде, швидше за все, мати стабільність у використанні.

Таблиця 1.1 - Переваги і недоліки архітектури «клієнт-сервер»

Системна характеристика	Значення
Переваги	
Мережа невеликих потужних машин	Якщо одна машина вийде з ладу, система все одно зможе продовжувати роботу
Потужні об'єднання комп'ютерів	Система надає потужність, що дозволяє виконувати роботу без монополізації ресурсів
Деякі робочі станції настільки ж потужні, як мейнфрейми, але їх вартість на порядок нижче	Надаючи обчислювальні потужності за менші гроші, система дозволяє вам витратити зекономлені кошти на інші придбання або на збільшення ваших доходів
Відкриті системи	Апаратуру, програми та послуги можна купувати у різних постачальників
Легкість нарощування системи	Систему неважко модернізувати, як тільки потреби зміняться
Індивідуальна робоче середовище клієнта	Об'єднання комп'ютерних платформ, підбираючи їх під конкретні потреби підрозділів і користувачів
Недоліки	

Слабка підтримка	Окремі частини системи не завжди коректно працюють разом
Недолік інструментальних засобів обслуговування	При використанні архітектури клієнт-сервер часто доводиться шукати інструментальні засоби на ринку або розробляти їх самостійно

Основна ідея архітектури «клієнт-сервер» полягає в поділі мережного додатку на кілька компонентів, кожен з яких реалізує специфічний набір сервісів. Компоненти такого додатка можуть виконуватися на різних комп'ютерах, виконуючи серверні і/або клієнтські функції. Це дозволяє підвищити надійність, безпеку і продуктивність мережеских додатків і мережі в цілому.

Застосування стратегії зумовлено двома причинами: інкапсуляція поведінки або алгоритму і можливість заміни поведінки під час виконання. Будь-який нормально спроектований клас вже інкапсулює в собі поведінку, але не будь-який клас з деяким поведінкою є або повинен бути стратегією. Стратегія потрібна тоді, коли важливо мати можливість заміни поведінок під час виконання. Іншими словами, стратегія забезпечує точку розширення системи.

Враховуючи можливість використання в програмі різних підходів для генерування дій розробки браузерної онлайн-гри на основі архітектури клієнт-сервер, для подальшого розвитку системи пропонується ввести блок для вибору параметрів роботи браузерної онлайн-гри.

1.5 Типи клієнт-серверних архітектур

Практичні реалізації клієнт-серверної архітектури називаються клієнт-серверними технологіями. Кожна технологія визначає власні або використовує наявні правила взаємодії між клієнтом і сервером, які називаються протоколом обміну (протоколом взаємодії).

					IT51.090БАК.002 ПЗ	Аркуш
						16
Зм.	Арк.	№ документа	Підпис	Дат		

Типи архітектур:

- Двокаскадна архітектура.
- Трьохкаскадна архітектура.
- Багатокаскадна архітектура.

1.6 Двокаскадна архітектура

У будь-якій мережі (навіть тимчасовій), побудованій на сучасних мережових технологіях, присутні елементи клієнт-серверного взаємодії, найчастіше на основі двохкаскадної архітектури. Двохкаскадна (two-tier, 2-tier) вона називається через необхідність розподілу трьох базових компонентів між двома вузлами (клієнтом і сервером), рисунок 1.3.

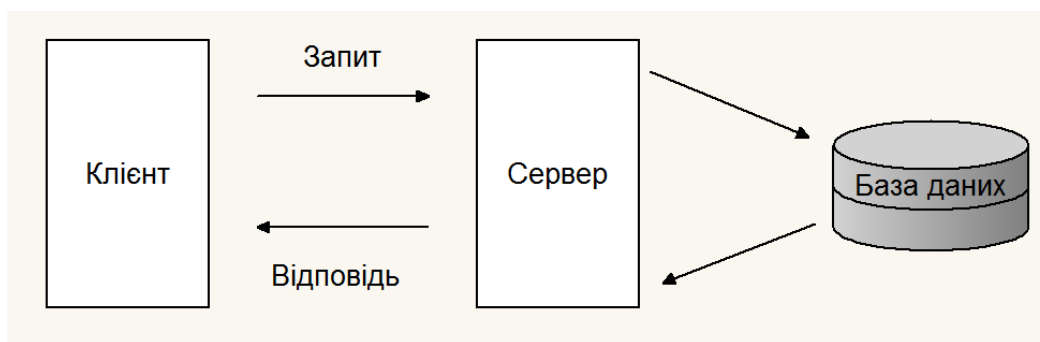


Рисунок 1.3 – Двохкаскадна архітектура

Двохкаскадна архітектура використовується в клієнт-серверних системах, де сервер відповідає на клієнтські запити безпосередньо і в повному обсязі, при цьому використовуючи тільки власні ресурси. Тобто сервер не викликає сторонні мережеві програми, але не звертається до сторонніх ресурсів для виконання будь-якої частини запиту

Розташування компонентів на стороні клієнта або сервера визначає наступні основні моделі їх взаємодії в рамках двухзвенной архітектури:

- Сервер терміналів - розподілене представлення даних.
- Файл-сервер - доступ до віддаленої бази даних і файлових ресурсів.
- Сервер БД - віддалене уявлення даних.

– Сервер додатків - віддалений додаток.

Перераховані моделі з варіаціями представлені на рисунку 1.4.

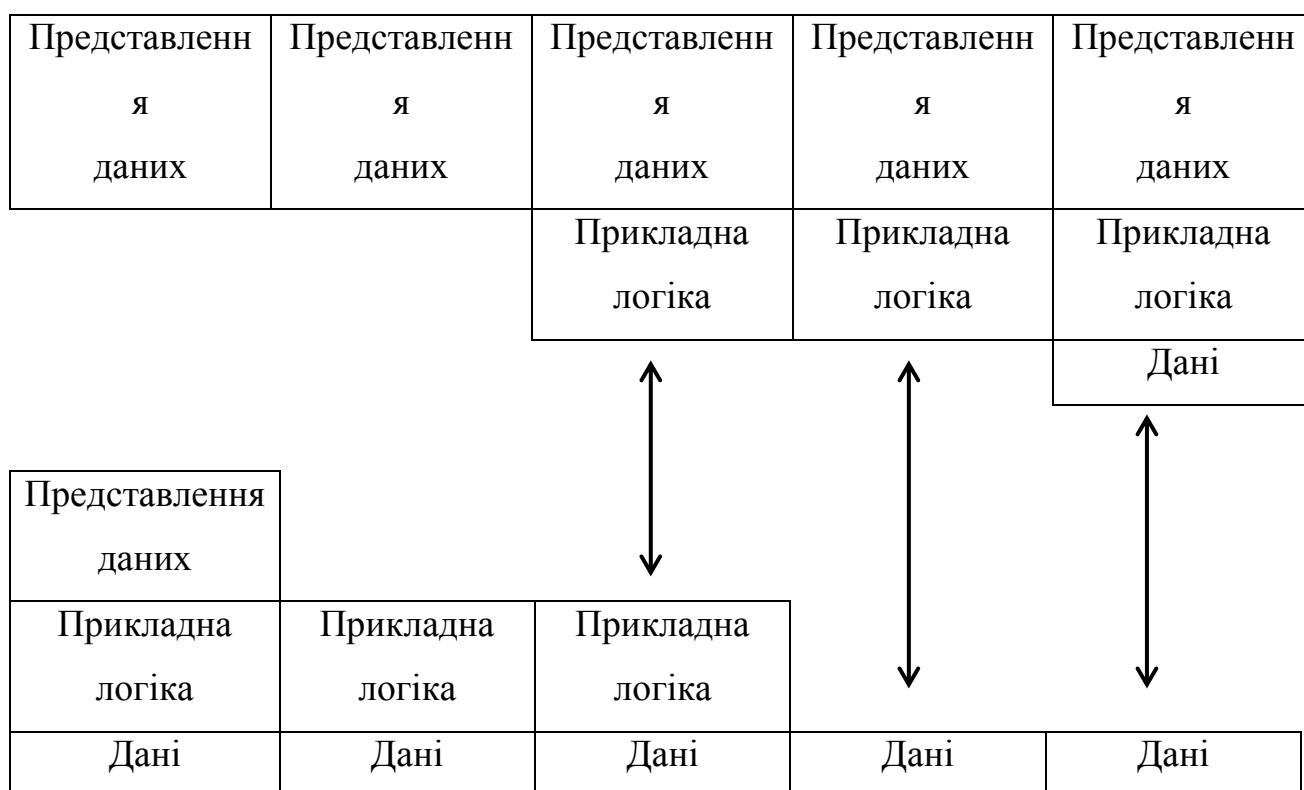


Рисунок 1.4 – Моделі клієнт-серверної взаємодії

З появою персональних комп'ютерів і локальних мереж, була реалізована модель файлового сервера, який представляв доступ файлових ресурсів до віддаленої бази даних. В цьому випадку виділений вузол мережі є файловим сервером, на якому розміщені файли бази даних. На клієнтах виконуються додатки, в яких поєднані компонент уявлення та прикладної компонент (СУБД і прикладна програма), що використовують підключену віддалену базу, як локальний файл. Протоколи обміну при цьому представляють набір низькорівневих викликів операційої файлової системи.

Така модель показала свою неефективність з огляду на те, що при активній роботі з таблицями БД виникає велике навантаження на мережу. Частковим вирішенням цієї проблеми є підтримка тиражування (реплікації) таблиць і запитів. В цьому випадку, наприклад при зміні даних, оновлюється не вся таблиця, а тільки модифікована її частина.

З появою спеціалізованих СУБД з'явилася можливість реалізації іншої моделі доступу до віддаленої бази даних - моделі сервера баз даних. В цьому випадку ядро СУБД функціонує на сервері, прикладна програма на клієнті, а протокол обміну забезпечується за допомогою мови SQL. Такий підхід в порівнянні з файловим сервером веде до зменшення завантаження мережі й уніфікації інтерфейсу “клієнт-сервер”. Однак, мережевий трафік залишається досить високим, крім того, як і раніше неможливо задовільний адміністрування додатків, оскільки в одній програмі поєднуються різні функції.

З розробкою і впровадженням на різні сервери баз даних механізму збережених процедур з'явилася концепція активного сервера БД. У цьому випадку частина функцій прикладного компонента реалізовані у вигляді збережених процедур, що виконуються на стороні сервера. Решта прикладна логіка виконується на стороні клієнта. Протокол взаємодії - відповідний діалект мови SQL.

Переваги такого підходу очевидні:

- Можливо централізоване адміністрування прикладних функцій.
- Зниження вартості володіння системою (TOC, total cost of ownership) за рахунок оренди сервера, а не його покупки.
- Значне пониження мережевого трафіку (тому що передаються не SQL-запити, а виклики збережених процедур).

Основний недолік - обмеженість коштів розробки збережених процедур у порівнянні з мовами високого рівня.

1.7 Трьохкаскадна архітектура

Ще одна тенденція в клієнт-серверних технологіях пов'язана з дедалі більшим використанням розподілених обчислень. Вони реалізуються на основі моделі сервера додатків, де мережевий додаток розділене на дві і більше частин, кожна з яких може виконуватися на окремому комп'ютері. Виділені частини

					IT51.090БАК.002 ПЗ	Аркуш
						19
Зм.	Арк.	№ документа	Підпис	Дат		

додатка взаємодіють один з одним, обмінюючись повідомленнями в заздалегідь узгодженому форматі. В цьому випадку двокаскадна клієнт-серверна архітектура стає трикаскадною (three-tier, 3-tier) рисунок 1.5.

Як правило, третім шаром в трьохкаскадній архітектурі стає сервер додатків, тобто компоненти розподіляються наступним чином:

- Представлення даних - на стороні клієнта.
- Прикладний компонент - на виділеному сервері додатків (як варіант, що виконує функції проміжного ПЗ).
- Управління ресурсами - на сервері БД, який і представляє запитувані дані.

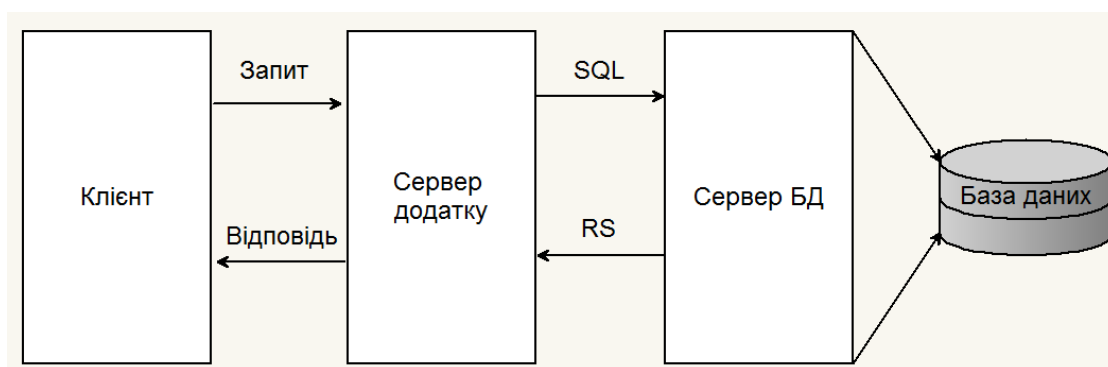


Рисунок 1.5 – Трьохкаскадна архітектура

1.8 Багатокаскадна архітектура

Трьохкаскадна архітектура може бути розширена до багатокаскадної (N-tier, Multi-tier) шляхом виділення додаткових серверів, кожен з яких представлятиме власні сервіси і користуватися послугами інших серверів різного рівня. Абстрактний приклад багатозіркової моделі наведено на рисунку 1.6.

Такі архітектури більш розумно розподіляти модулями обробки даних, які в цьому випадку виконуються на одному або декількох окремих серверах. Ці програмні модулі виконують функції сервера для інтерфейсів з користувачами і клієнтами - для серверів баз даних. Крім того, різні сервери додатків можуть взаємодіяти між собою для більш точного поділу системи на функціональні блоки, які виконують певні ролі. Наприклад, можна виділити сервер управління

персоналом, який буде виконувати всі необхідні для управління персоналом функції. Зв'язавши з ним окрему базу даних, можна приховати від користувачів всі деталі реалізації цього сервера, дозволивши їм звертатися тільки до його загальнодоступним функцій. Крім того, таку систему дуже просто адаптувати до Web, оскільки простіше розробити html-форми для доступу користувачів до певних функцій бази даних, ніж до всіх даних.

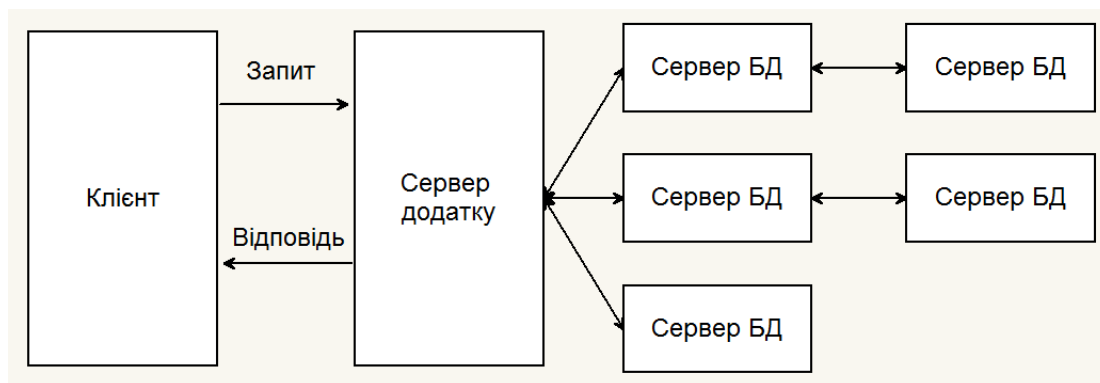


Рисунок 1.6 – Багатокаскадна архітектура

1.9 Порівняння архітектур

Двохкаскадна архітектура простіше, так як всі запити обслуговуються одним сервером, але саме через це вона менш надійна і висуває підвищені вимоги до продуктивності сервера.

Трьохкаскадна архітектура складніше, але завдяки тому, що функції розподілені між серверами другого і третього рівня, ця архітектура являє:

- Високий ступінь гнучкості і масштабованості.
- Високу безпеку (тому що захист можна визначити для кожного сервісу або рівня).
- Високу продуктивність (тому що завдання розподілені між серверами).

1.10 Клієнт-серверні технології

Архітектура клієнт-сервер застосовується у великій кількості мережових технологій, що використовуються для доступу до різних мережових сервісів. Коротко розглянемо деякі типи таких сервісів (і серверів).

Web-сервери - спочатку представляли доступ до гіпертекстових документів по протоколу HTTP (Hyper Text Transfer Protocol). Зараз підтримують розширені можливості, зокрема роботу з бінарними файлами (зображення, мультимедіа тощо).

Сервери додатків - призначені для централізованого вирішення прикладних завдань в деякій предметній області. Для цього користувачі мають право запускати серверні програми на виконання. Використання серверів додатків дозволяє знизити вимоги до конфігурації клієнтів і спрощує загальне управління мережею.

Сервери баз даних - сервери баз даних використовуються для обробки запитів користувачів на мові SQL. При цьому СУБД знаходиться на сервері, до якого і підключаються клієнтські програми.

Файл-сервери - зберігають інформацію у вигляді файлів і представляє користувачам доступ до неї. Як правило файл-сервер забезпечує і певний рівень захисту від несанкціонованого доступу.

Проксі-сервер - по-перше, діє як посередник, допомагаючи користувачам отримати інформацію з Інтернету і при цьому забезпечуючи захист мережі. По-друге, зберігає часто запитувану інформацію в кеш-пам'яті на локальному диску, швидко доставляючи її користувачам без повторного звернення до Інтернету.

Файрволи (брандмауери) - міжмережові екрани, що аналізують і фільтрують проходить мережовий трафік, з метою забезпечення безпеки мережі.

Поштові сервери - надають послуги з відправлення та одержання електронних поштових повідомлень.

					IT51.090БАК.002 ПЗ	Аркуш
						22
Зм.	Арк.	№ документа	Підпис	Дат		

Сервери віддаленого доступу (RAS) - ці системи забезпечують зв'язок з мережею по комутованих лініях. Віддалений співробітник може використовувати ресурси корпоративної ЛВС, підключившись до неї за допомогою звичайного модему.

Це лише кілька типів зі всієї великої кількості клієнт-серверних технологій, що використовуються як в локальних, так і в глобальних мережах.

Висновки до розділу

У роботі досліджено особливості функціонування архітектур клієнт-сервер. Аналіз містить аналіз типів архітектур клієнт-серверів.

В цьому розділі були розглянуті основні класи клієнт-серверних архітектур, типи архітектур клієнт-серверних додатків, їх переваги та недоліки.

Проаналізувавши переваги та недоліки клієнт-серверних архітектур, можна зробити висновок, що для браузерної онлайн-гри є доцільним обрання тришарової архітектури, як каркасу додатка. Збільшення потужності клієнтських машин дозволяє реалізувати роботу системи в рамках моделі клієнт-сервер з «товстим» клієнтом, що відповідає сучасним тенденціям у галузі веб-додатків. Важливим аспектом є обрання типу взаємодії і з огляду на переваги та недоліки вищеповисаних типів взаємодії, буде обрано архітектурний стиль REST, як швидкий для вивчення та легший для реалізації і взаємодії. Наступний розділ матиме за мету пояснити актуальність створення нової системи та підібрати інструменти для реалізації.

					IT51.090БАК.002 ПЗ	Аркуш
						23
Зм.	Арк.	№ документа	Підпис	Дат		

2. ОГЛЯД ІСНУЮЧИХ ПРОГРАМНИХ РІШЕНЬ ТА ТЕХНОЛОГІЙ

У процесі розробки головний етап це визначення та обґрунтування вибраних технологій, з якими буде працювати розробник системи, тому в цьому розділі буде пояснено, що саме було вибрано для розробки програмного забезпечення.

2.1 Браузерна онлайн-гра загальні відомості

Серед безлічі категорій комп'ютерних ігор браузерні онлайн-ігри відрізняються своєю незалежністю від комп'ютерних платформ і встановлених на них операційних систем(ОС). Також, грати в браузерні ММО ігри можна без завантаження і установки додаткового програмного забезпечення, що дозволить продовжити розпочату гру з будь-якого комп'ютерного пристрою з встановленим браузером, підключеного до Інтернету.

Браузерні онлайн-ігри, які використовують інтерфейс браузера і не вимагають установки на ПК додаткових додатків поділяються на три види , що зображено на рисунку 2.1.

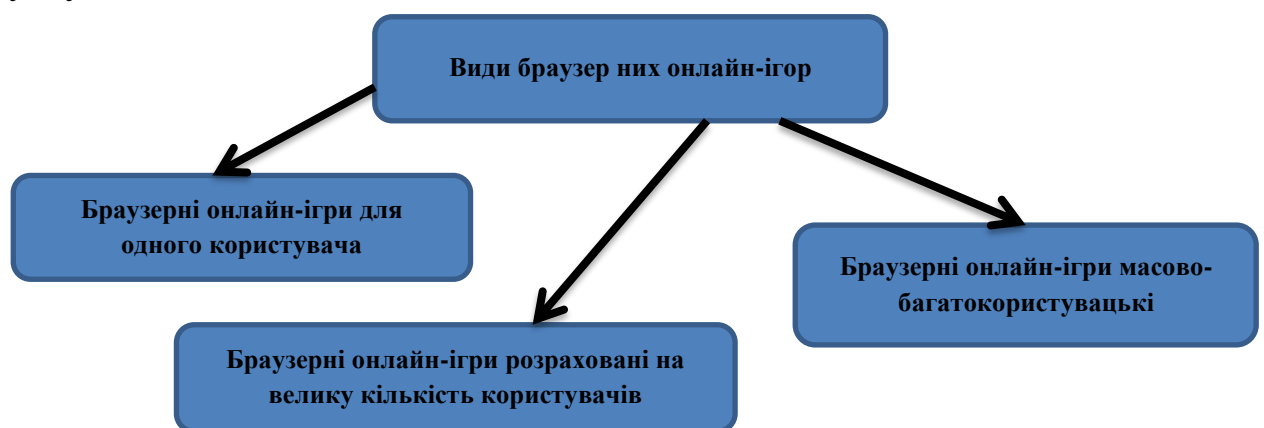


Рисунок 2.1 – Види браузерних онлайн-ігор

2.2 Браузерні онлайн-ігри для одного користувача

Найпоширенішими серед браузерних онлайн-ігор на одного користувача є флеш гри. До того, як технологія створення flash-ігор отримала широке поширення, активно використовувалася технологія java.

Поодинокі браузерні онлайн-ігри легко освоїти навіть недосвідченому користувачу інтернет. Характерною особливістю таких браузерних онлайн-ігор є неможливість збереження результатів, ігрової позиції, а тому кожен раз ігровий процес доводиться починати спочатку. Найчастіше браузерні онлайн-ігри на одного користувача зараховують до ігор - казуальних.

2.3 Браузерні онлайн-ігри розраховані на велику кількість користувачів

Як приклад ігор багатокористувацьких можна привести ті, які імітують настільні ігри: шахи, шашки, нарди, ігри в карти, що зображено на рисунку 2.2. Є можливість браузерної онлайн-гри удвох, як є і сеанси браузерної онлайн-гри з декількома партнерами. Браузер допомагає забезпечити зв'язок між гравцями за допомогою сервера. Ігровий процес може вимагати одноразової присутності всіх учасників партії, так і ця умова може бути необов'язковим. Якщо присутність онлайн не обов'язково, то ходи зберігає ігровий сервер.

					IT51.090БАК.002 ПЗ	Аркуш
						25
Зм.	Арк.	№ документа	Підпис	Дат		



Рисунок 2.2 – Багатокористувацька онлайн-гра, яка імітує настільну гру[1]

2.4 Браузерні онлайн-ігри масово-багатокористувацькі

Такі браузерні онлайн-ігри характерні тим, що забезпечують взаємодію сотням, тисячам гравцям одночасно. У більшості браузерних масово-багатокористувацьких ігор інтерфейс виконаний за допомогою HTML, у вигляді сторінок із зображеннями і текстом.

Мінімальну інтерактивність забезпечують за допомогою JavaScript, а також flash технологій, які дозволяють браузерній онлайн-грі стати набагато привабливіше зовні. Велика кількість анімації, на жаль, може привести до того, що браузерна онлайн-гра буде перевантажена і почне пригальмовувати.

Найчастіше браузерні онлайн-ігри масово-багатокористувацькі розробляють в жанрі RPG, так само як стратегію або гру економічну. До масово-багатокористувацьких браузерних онлайн-ігор також зараховують сайти з віртуальними «тваринами». Процес гри буває так цікавий, що розтягується на довгі роки.

Є браузерні онлайн-ігри, які мають на увазі покупку артефактів, додаткових речей або зброї за реальні гроші, тим самим сильніше залучаючи користувачів в ігровий процес. Нерідко браузерні онлайн ігри мають наявність реклами.

2.5 Порівняння жанрів браузерних онлайн-ігр

Найпростіший спосіб досягти успіху – подивитися як це роблять інші, взяти існуючі ідеї і відшліфувати їх до блиску. Розвитком цього етапу займається сценаристи і дизайнери. Загальну тему для сценарію браузерної онлайн-гри можна взяти з таких тем як:

- Фантастика: майбутнє, космос.
- Фентезі, міфологія, релігія.
- Реальність, історія, кримінал, війна, симуляція, будівництво, спорт.
- Математичні та фізичні головоломки.
- Художня література.

Далі – продумати проти чого користувач повинен боротися і чим керувати. Це конкретні активні користувачі(противники) або ж результат діяльності самого користувача (логічні ігри). В результаті сценарію браузерна онлайн-гра так чи інакше призводить до жанру і способу управління:

– Аркади. Як правило легко реалізовані, двовимірного простору з простими правилами зіткнень та рухів. Це можуть бути космічні «стріляли»: на flash Tennis, Snake, Pacman, Bomberman, Mario.

– Квести та пригоди. До теперішнього часу практично тип перестав існувати як вид, втративши свою присутність у RPG. Розвивається індуктивність гравця по складанню і поєднанню речей для досягнення мети: Monkey island, sam & max, grim fandango.

– 2х і 2.5-мірні (ізометричні) стратегії та «рольові» ігри. Тренують у користувача баланс ресурсів і виконання квестів відповідно: Dune, Warcraft, C & C.

					IT51.090БАК.002 ПЗ	Аркуш
						27
Зм.	Арк.	№ документа	Підпис	Дат		

– 2.5D і 3D онлайн-ігри. Браузерні онлайн-ігри на реакцію і планові витрата патронів. Квести як правило надто спрощені.

– Нескінченні будови. Браузерні онлайн-ігри теоретично нескінченні типу: SimCity, Sims, Black and White, Populous. Мета в підтримці процесу за рахунок керування численними падаючими параметрами злочинності, пожежонебезпеки, голоду, сну та інших.

– ММО – в загальному випадку не обов'язково RPG, тут основна мета користувача – здійснювати дії на пару з іншими користувачами, а мета розробника – збільшення фінансових ресурсів за рахунок кількості користувачів: «Бійцівський клуб», Ферма.

Прогортавши каталог браузерних онлайн-ігор, можна прослідкувати безліч онлайн-ігор в жанрі ММОРПГ.

По-перше, це створювач всіх браузерних онлайн-ігор. Тому такі проекти, як World of Warcraft і Lineage сприяли просуванню багатокористувацьких проектів на широкий ринок. Досвідчені користувачі відносять ці ігри до «легенд». Свого часу браузерні онлайн-ігри викликали справжній фурор серед користувачів, дозволяючи відкрито досліджувати локації і створювати персонажів по індивідуальному образу. При цьому персонаж необхідно було розвивати, а масштабні сутички з сильними персонажами разом з товаришами залишаться в пам'яті користувачів надовго.

По-друге, ММОРПГ відкривають великі можливості для користувачів. Сьогодні «рольові» елементи з'являються і в інших жанрах. Якщо раніше дія відбувалася лише одній локації, то зараз з'явилась можливість потрапити у декілька локацій. Розробники створюють різноманітні ситуації - наприклад, можливість опинитися в майбутньому (Star Conflict) або далекому минулому (ARK: Survival Evolved), битися з гігантськими вигаданими монстрами або бандитами. Присутні «рольові елементи», такі як: прокачка героя, кастомізація, дослідження під земель і інше.

					IT51.090БАК.002 ПЗ	Аркуш
						28
Зм.	Арк.	№ документа	Підпис	Дат		

По-третє, масштабні дії. Так, зявилась можливість відчувати збільшення кількості динаміки, але в ММОРПГ відтворюються дійсно масштабні події: між серверні битви, облоги замків, кланові війни. Кількість користувачів може перевищувати сотні учасників, і часом сутички тривають по кілька реальних днів. На одній території може перебувати тисячі унікальних персонажів з неймовірними здібностями, за допомогою яких можна змінити хід бою. Також все залежить від тактики і командної роботи.

Світ браузерних онлайн-ігор постійно розвивається – з'являються нові проекти з використанням останніх технологій, а креативні розробники намагаються розробити незвичайні елементи в ігровий процес. Так створюється формула ідеальної гри, і в деяких ігрових індустріях відмінно реалізуються: Blade & Soul, Black Desert, World of Warcraft.

2.6 Огляд браузерних онлайн-ігор жанру ММОРПГ

Існує одно користувачький, типовий представник сімейства флеш-гри розрахований на одного користувача, який вже знайомий абсолютно всім, від малих до великих. Як правило, такі браузерні онлайн-ігри дуже прості в освоєнні і не вимагають від користувача рішення складних математичних або логічних задач. Вони відмінно підходять, для того щоб скоротати час.

Далі йдуть браузерні онлайн-ігри на велику кількість користувачів, які також знайомі багатьом, любителям весело проводити час. До таких ігор можна віднести онлайн шахи та інші схожі проекти. У багатокористувачьких іграх гра відбувається з справжнім супротивником, який також думає і може діяти не стандартно.

Drakensang Online. Світ «Дракенсанг» online – Дорія, величезний на даний момент, охолює 10 регіонів, кожен з оригінальними розробками, особливостями та історією, причому організована глобальна історична прив'язка. Природно потрібно запастися як зброєю, обладунками, так іншими диво провіантом.

					IT51.090БАК.002 ПЗ	Аркуш
						29
Зм.	Арк.	№ документа	Підпис	Дат		

Переваги:

- Величезна кількість завдань оперізають епічну сагу.
- Приємна 3D графіка і супер ефекти.
- Рідкісні спеціальні навички.
- Різноманітність місцевості для подорожей, заселена ворожими мешканцями.
- Одночасно зануритися в процес з тисячами геймерами.

Трепетний момент, вибір клас персонажа, поки три користувача, скажімо класичний «джентльменський набір»:

- Маги кола.
- Війни дракона.
- Слідопит.

Розбрат. Ігровий процес відбувається безпосередньо в Real time, позначається з наявності технології Flash. Браузерна онлайн-гра «Розбрат» виконана в популярному жанрі MMORPG.

Користувачів очікує:

- Непогана графіка і опрацювання внутрішнього світу.
- Цікавий розвиток історії персонажа, забезпечується розвиненою побудовою талантів.

- Велика сфера професій.
- В наявності бойові вихованці.
- Сотні квестів і всіляких завдань.
- За виконання останніх видаються призи.
- Створення і розвиток власної гільдії.
- Групові бої і випадкові події.
- Створювання до 4 неповторних героїв з декількох класів.

У «Розбрат» чотири раси: зелені, довговухі, накачані і білі.

Раси:

- Орки – люті вояки, яких так виховують з дитинства.

					IT51.090БАК.002 ПЗ	Аркуш
						30
Зм.	Арк.	№ документа	Підпис	Дат		

- Ельф – загадкова, горда і своєрідна народність, визнаючи своїх богів.
- Гноми – не дружать з магією, але виготовляють під землею зброю, броню і технічні пристрої, які називаються чудесами інженерної думки.

- Люди – люблять подорожувати і освоювати всілякі професії, шуміти і веселитися.

Класи:

- Мисливець (тільки орки) – гостре око, відданий звір-друг, що ще треба для злиття з природою і полювання.

- Шаман (Зеленокожі упирі) - заклик тотемів на поле бою.

- Воїн (люди і Green гади) - серце бою, в гущі подій і пеклі, вони надихають інших йти на подвиги.

- Містик (Only ельфи) – досконало володіють магичною зброєю, приборкують воду.

- Маг (люди) - здоров'я не фонтан, але володіє потужною стихією в битві.

- Жрець (знову Human) - гідності: зв'язок з богами духовність і віра, цікавий суб'єкт, може друзів зцілювати.

- Берсерк (тільки гноми) – експерти по володінню металевими сокирами і мечами як одnorучного, так і дворучними.

Грані Реальності. Браузерна онлайн-гра «Грані Реальності» унікальна онлайн-гра з величезним потенціалом.

Грані Реальності – «рольова» багатокористувацька браузерна онлайн-гра – ММОРПГ, що залежить від взаємодії з користувачами в режимі реального часу – брати участь в боях, кланових війнах, групових та індивідуальних протистояннях, а також в торгових угодах.

2.7 Технологія WebAssembly

Завдяки новій технології, може змінитися то, як люди використовують комп'ютерні пристрої і інтернет.

					IT51.090БАК.002 ПЗ	Аркуш
						31
Зм.	Арк.	№ документа	Підпис	Дат		

Технологія WebAssembly є бібліотекою для компіляції програмного коду додатків в бінарний код формату Асемблера, який може бути запущений в браузері. На відміну від Flash технології (яка може бути скасована в недалекому майбутньому), підтримка WebAssembly вбудована в браузери також як і підтримка JavaScript, так що для його роботи ніяких плагінів встановлювати не знадобиться. На рисунку 2.3 зображено процес технології WebAssembly «від вихідного коду до Інтернету».

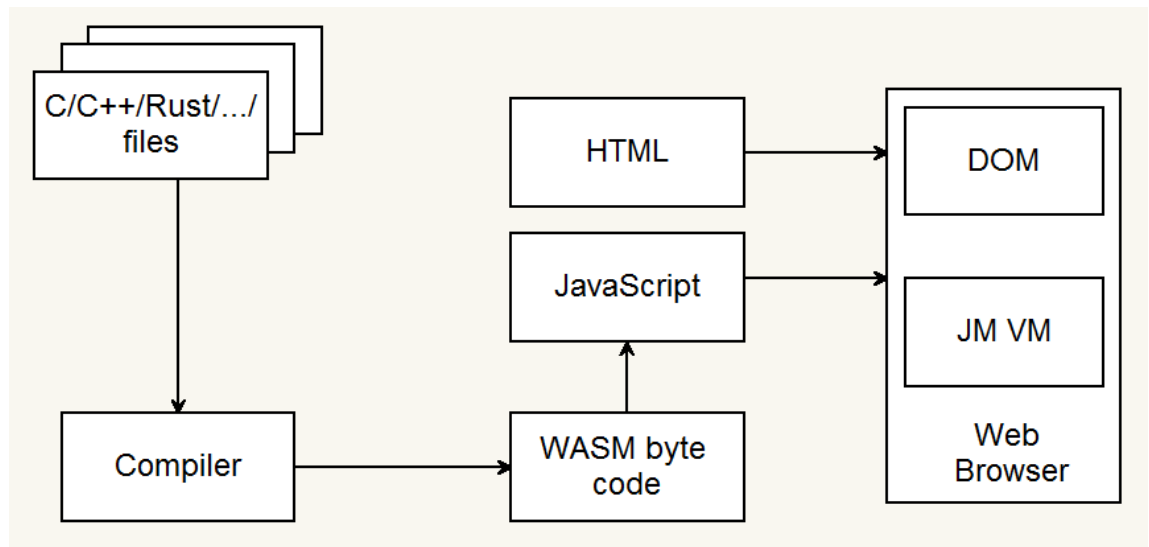


Рисунок 2.3 – процес технології WebAssembly «від вихідного коду до Інтернету»[2]

Будь-яка десктопна програма являє собою рядки, що складаються тільки з одиниць і нулів, які розуміє комп'ютер. Такий формат називається бінарним. Бінарні програми створюються шляхом компіляції додатків, написаних програмістами на одній з мов програмування. Але так як інший персональний комп'ютер(ПК) не розуміє мов програмування, програму доводиться компілювати в бінарний формат - в нулі та одиниці.

WebAssembly - бінарна мова Асемблера, також складається з нулів і одиниць, що сприймаються будь-яким комп'ютером. Таким чином, комп'ютерній мережі не будуть потрібні додаткові ресурси для компіляції цього коду, що відіб'ється на високій продуктивності додатка.

За допомогою WebAssembly програми можна запускати в браузері на швидкості порівнянної зі швидкістю роботи звичайної програми.

Так як WebAssembly є бінарний код, який запускається на локальній машині кінцевого користувача, робота таких додатків в браузері як онлайн-гра буде швидшою.

2.8 Технологія HTML5

Налаштувати середовище розробки HTML5 для розробника досить просто. Ця платформа не має залежності до IDE (інструмент ппортування написаної браузерної онлайн-гри на різні ігрові платформи, такі як iPad, Android, консолі, ПК, браузери).

Не дивлячись на легкість доступу до самих різних гаджетів, в HTML5 буде досить складно спочатку налаштувати індивідуальний ігровий проект. Також, при виборі цієї платформи для створення браузерної онлайн-гри знадобиться встановити локальний сервер.

Технологія HTML5 відмінно підходить для створення нескладних казуальних ігор. JavaScript, на якому створюється код розробником, є простий. Більш того, безліч модулів доповнюють цю мову програмування та безліч бібліотек є у відкритому доступі. Серед них можна відзначити такі компілятори, як CoffeeScript і TypeScript.

Величезний вибір доступних бібліотек можна назвати і негативною стороною вибору технології HTML5 для розробки браузерної онлайн-гри. Багато з них знаходяться ще в процесі розробки і можуть працювати нестабільно. Серед інших мінусів інструментів HTML5 виділяється їх обмеження в інтеграції, труднощі у створенні своїх бібліотек і погану продуктивність самої технології.

Так як HTML5 є веб-інтерфейсом, то видати написану на ньому онлайн-гру в Інтернеті для браузерів комп'ютерів і мобільних телефонів буде не складно. І

					IT51.090БАК.002 ПЗ	Аркуш
						33
Зм.	Арк.	№ документа	Підпис	Дат		

ніякого додаткового плагіна для портування браузерної онлайн-гри на різні платформи не буде потрібно.

Незважаючи на такі зручності, видання браузерної онлайн-гри на HTML5 у вигляді офлайн-версії ускладнене тим, що така можливість опрацьована ще не до кінця. На рисунку 2.4 зображено переваги технології HTML5.

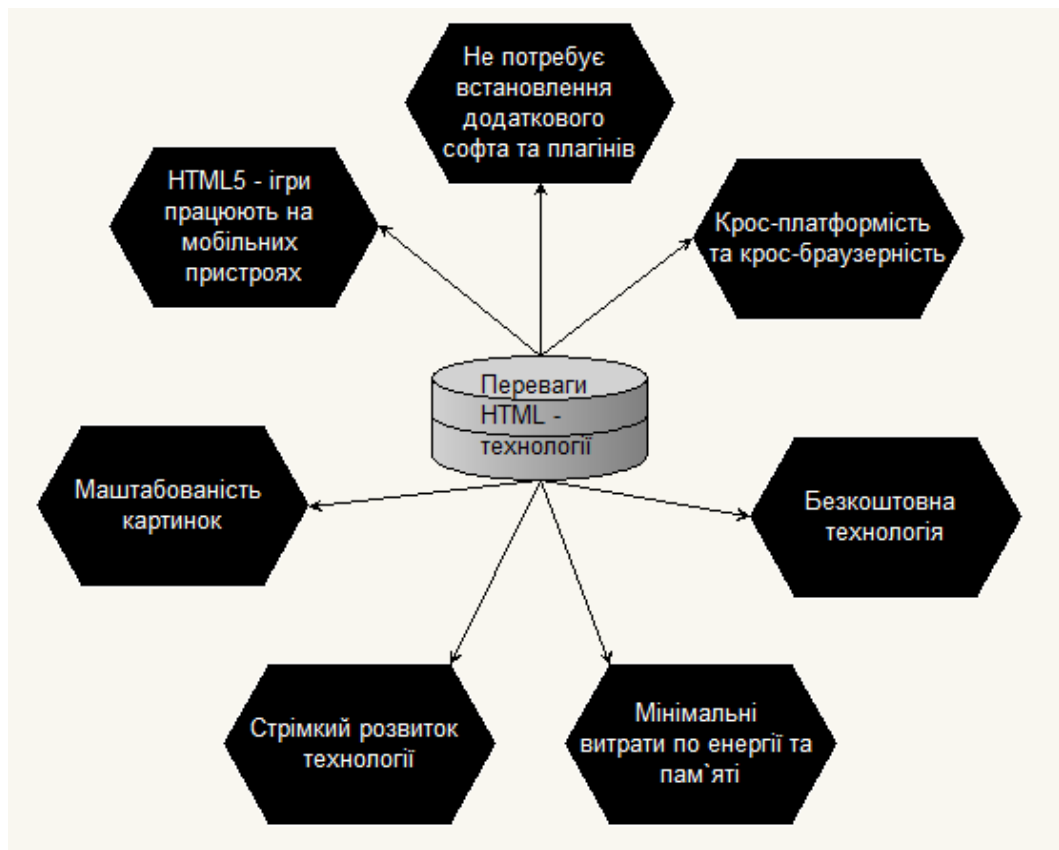


Рисунок 2.4 – Переваги технології HTML5

2.9 Технологія Unity3D

Технологія Unity може бути корисна розробнику автономного движка. Unity – технологію характеризують, як платформу, що реалізує дії в один клік. Це стосується запуску самого редактора, підключення різних середовищ розробки, тестування браузерної онлайн-гри.

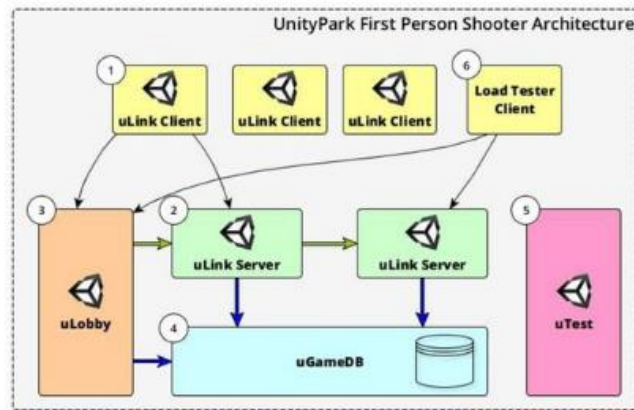


Рисунок 2.5 – Технологія Unity3D[3]

Мінусом Unity для розробників визначається приналежність технології до однієї компанії. Вихідні тексти цього движка не відкриті, і якщо в ньому знайдеться будь-які помилки, то виправити їх самотійно не має змоги. В цьому випадку доведеться скласти звернення спеціальній групі Beta-Team, яка постарается оперативно виправити знайдений недолік.

Другий мінус технології Unity, це вимушене використання вбудованого в середовище розробки IDE. Такі обмеження ввела компанія Unity3D і обійти їх не вдасться.

Для розробки браузерних онлайн-ігор технологія Unity, підходить ідеально. Програмна частина технології (движок) спочатку розроблявся для створення 2D і 3D браузерних онлайн-ігор. В технології є:

- Вбудована система моделювання фізики.
- Попередній перегляд роботи коду.
- Можливість складання окремих компонентів.
- Простота в роботі з інтерфейсом.

Проблемою створення браузерних онлайн-ігор на Unity можна назвати прив'язаність програмної частини до єдиної моделі створення ігор. І якщо розробник має своє бачення ігрового процесу, йому буде складно реалізувати свої ідеї в цьому редакторі. Також, мова C#, на якому пишеться код в Unity, може дещо ускладнити логіку браузерної онлайн-гри.

Плюс технології Unity в виданні браузерних онлайн-ігор - це простота їх портування. Щоб перенести створену браузерну онлайн-гру на ігрову консоль або мобільний телефон, досить натиснути одну кнопку. При цьому, створені на Unity браузерні онлайн-ігри відрізняються високою продуктивністю навіть на апаратах з недостатньою кількістю ресурсів.

Простота видання браузерних онлайн-ігор на Unity покладає на розробника фінансові турботи. Для включення можливості портування гри за кожен окрему платформу доведеться платити. Цей фактор обмежує кількість розробників, бажаючих працювати з даною технологією. Спільно з цим, навіть зараз, коли технологія Unity3D знаходиться на піку своєї популярності, вона встановлена, приблизно, на 80% апаратів потенційних користувачів.

2.10 Компоненти браузерної онлайн-гри

В першу чергу, має бути присутня інтрига, здатна збуджувати уяву потенційних гравців, незалежно від статі і віку. По-друге, повинні мати місце вічні людські цінності, оспівані класиками і актуальні понині-любов, дружба, і, бажано, можливість створення сімейних відносин. Однак навіть це не гарантує те, що створення браузерної онлайн-гри окупиться.

Створення онлайн ігри це ні в якому разі не шахи і не шашки, втілені в інтерактивному варіанті. Браузерні онлайн-ігри, що вимагають розумового процесу, затребувані дуже вузьким колом людей. Куди більшою популярністю серед користувачів різного віку і поколінь користуються браузерні онлайн-ігри, де герой, створений гравцем, б'ється, переходячи від одного бою до іншого.

					<i>IT51.090БАК.002 ПЗ</i>	Аркуш
						36
Зм.	Арк.	№ документа	Підпис	Дат		

Створення браузерної онлайн-гри передбачає обов'язкову умову: відкритий фінал. Іншими словами, гра може тривати нескінченно, інакше, знищивши ворога, користувач просто піде на інший ресурс. Важливо постійно доповнювати вже створену браузерну онлайн-гру новими модулями: локаціями, а так же персонажами, знищуючи яких користувач отримує такий бажаний досвід. Важливо при створенні браузерної онлайн-гри залишити користувачам можливість взаємодіяти один з одним. Вбивати намальованих розробниками монстрів можна в будь-якій грі, але саме вбивати супротивника, знаючи, що героєм на екрані управляє інша людина, цікавіше в два рази.

Створення браузерної онлайн-гри - це в першу чергу можливість взаємодії, а тому створення коаліцій, об'єднань, кланів, найбільш цікавить в ігровому процесі потенційних користувачів. Можливість об'єднатися в групу це схоже на стадний інстинкту, адже в об'єднанні людей набагато простіше вижити, як в реальному світі, так і в ігровому. Крім того, якщо у кого-небудь в гільдії з'являється новий обладунок, куплений додатково, то через деякий час хтось замовить собі в точності такий же.

При створенні браузерної онлайн-гри важливо пам'ятати і про грошовий фактор. Більшість користувачів ніколи не упустять можливості «поліпшити» свого героя, подарувавши йому більш міцну броню, більш нищівний меч, сильне бойове або їдова тварина, а так само можливість плавати або літати, зрозуміло, заплативши додатково.

Створення браузерних онлайн-ігор це не тільки творчий процес, що має на увазі постійні новаторські ідеї, а й постійно розвивається прибутковий бізнес.

Висновки до розділу

Даний розділ присвячений огляду та порівнянню існуючих рішень, як з точки зору інструментів розробника для створення браузерних онлайн-ігор, так і існуючих систем на ринку розробників та корпорацій. Проведено стислий огляд та аналіз технологій розроблення та впровадження браузерних онлайн-ігор. На

					<i>IT51.090БАК.002 ПЗ</i>	Аркуш
						37
Зм.	Арк.	№ документа	Підпис	Дат		

основі нього можна зробити висновок про наявність потреби створення рішення, яке буде задовольняти потреби користувачів. Реалізація повинна ввібрати максимум найкращих рис з розглянутих технологій, а саме простий, зрозумілий інтерфейс, легкість в опануванні та потрібна кількість ресурсів застосування. У цьому розділі було описано основні технології: WebAssembly, HTML5 та Unity3D, що дозволяє якісно розроблювати браузерні онлайн-ігри різних видів, зменшуючи кількість написаного коду та ресурсоемності. До того наявність широкого спектру модулів, що здатні задовольнити майже будь-які потреби як розробника, так і браузерної онлайн-гри.

					IT51.090БАК.002 ПЗ	Аркуш
						38
Зм.	Арк.	№ документа	Підпис	Дат		

3. ПРОГРАМНА РЕАЛІЗАЦІЯ ДОДАТКУ

3.1 HTML5 і Node.js: багатокористувацька браузерна онлайн-гра на архітектурі клієнт-сервер

В даній роботі використано MMORPG на основі браузера. Backend написаний у Node.js і використовує MongoDB для збереження даних карти. Frontend використовує тег HTML5 Canvas, щоб обробляти малюнок на карті, і карта розміщується, коли сторінка завантажується відповідно до розміру екрана користувачів. Також використовується аудіо HTML5 для звукових ефектів. На рисунку 3.1 – зображено файли розробки браузерної онлайн-гри

assets	Updating nametags more frequently
lib	Simpler map format. BREAKS EXISTING GAMES
resources	Simpler map format. BREAKS EXISTING GAMES
.gitignore	removing node_modules from repo. Run 'npm install' for now on.
README.md	updating readme
favicon.ico	Adding favicon.ico file
index.html	Upgrading jQuery from 1.7.1 to 2.1.0.
map.json	Simpler map format. BREAKS EXISTING GAMES
package.json	version bump
server.js	node server.js PORT=80 INTERFACE=0.0.0.0

Рисунок 3.1 – Файли розробки браузерної онлайн-гри

```
@@ -19,7 +19,7 @@ var npcs      = require('./lib/npccs.js').setMap(map).setSocket(io).setPlayers(  
  
// Web Server Configuration  
var server_port = parseInt(process.argv[2], 10) || 80; // most OS's will require sudo to listen on 80  
- var server_host = null;  
+ var server_host = process.argv[3] || null;  
  
var mongo_connection_string = 'mongodb://127.0.0.1:27017/terraformia';
```

Рисунок 3.2 – Використання сервера в програмному коді

					IT51.090БАК.002 ПЗ	Аркуш
						39
Зм.	Арк.	№ документа	Підпис	Дат		

```
@@ -1,6 +1,6 @@
{
    "name": "Cobalt-Calibur-3",
-   "version": "0.3.0",
+   "version": "0.4.0",
    "author": "Thomas Hunter <me@thomashunter.name>",
    "description": "2D MMO Engine",
    "repository": {

```

Рисунок 3.3 – Використання версії «bump» програмному коді

3.2 Процес браузерної онлайн-гра на архітектурі клієнт-сервер

Рухи виконуються за допомогою клавіш «WASD» та стрілок. При натисканні клавіш 1-6 виконується побудова вказаних об'єктів. Натискання клавіші F стимулює функцію «Дія з об'єктом». Збираючи об'єкти, інвентар-ресурси у верхньому правому куті збільшуються. Коли розміщуються користувачем об'єкти, ресурси відповідно по кількості - зменшуються. На відміну від інших ігор (наприклад, Minecraft), не зберігається створені в інвентарі елементи, а лише сировина.

```
1 -> Wooden Wall (4 wood)
2 -> Wooden Floor (2 wood)
3 -> Stone Wall (4 stone)
4 -> Stone Floor (2 stone)
5 -> Wooden Door (12 wood)
6 -> Glass Window (4 sand)
```

Рисунок 3.4 – Синтетичні об'єкти

Розміщення синтетичних об'єктів призведе до повільного виштовхування інших користувачі браузерної онлайн-гри на архітектурі клієнт-сервер. «Об'єкти»

					IT51.090БАК.002 ПЗ	Аркуш
						40
Зм.	Арк.	№ документа	Підпис	Дат		

будуть блокувати користувачів-ворогів, але не будуть блокувати користувачів-гравців.

```
Trees -> Mine to receive 2 wood, becomes stump
Stump -> Mine to receive 1 wood, becomes grass
Grass -> Walk on it. Randomly becomes tree
Dirt -> Walk on it. Randomly becomes grass
Big Ore -> Mine to receive 2 ore, becomes small ore
Small Ore -> Mine to receive 1 ore, becomes rubble
Big Stone -> Mine to receive 2 stone, becomes small stone
Small Stone -> Mine to receive 1 stone, becomes rubble
Rubble -> Mine to receive 1 stone, becomes dirt
Water -> Makes sand from adjacent ground
```

Рисунок 3.5 – Натуральні об'єкти

В «день» трава може вирощувати дерево, бруд може вирощувати траву. Бруд і трава, що прилягає до води, стануть піском. Коли відбувається землетрус (кожні кілька днів), він викидає в світ більше каменю і руди (і, можливо, пошкодить будівлі).

Коли сервер запускається, NPC можуть з'являтися на природних об'єктах.

Є денні та нічні цикли, але вони не впливають на «геймплей». Є випадкові землетруси, і вони змінюють ландшафт. Ресурс «життя» повільно відновлюється.

На рисунку 3.6 зображено знімок екрана (нічний час).



Рисунок 3.6 – Зміна часу – ніч у браузерній онлайн-грі на архітектурі клієнт-сервер

					IT51.090БАК.002 ПЗ	Аркуш
						41
Зм.	Арк.	№ документа	Підпис	Дат		

Написане крок за кроком керівництво, як завантажити Cobalt Calibur, і отримання всіх хостингів на платформі Red Hat в OpenShift. Це включає налаштування програми Node.js і налаштування MongoDB. Найкраща частина полягає в тому, що це повністю безкоштовно та займає 30 хвилин.

3.3 Діаграма варіантів використання для програмного забезпечення браузерної онлайн-гра на архітектурі клієнт-сервер

Основні варіанти використання:

Головний актор: Користувач

Рівень взаємодії: Користувач – Система

Передумова: Користувач почав вхід до системи.

Основний сценарій:

Користувач створює об'єкти.

Кнопка «Створити об'єкт».

Користувач виконує рухи.

Кнопки «Вправо, вліво, вперед, назад».

Вкладка «Виконання дії з об'єктами».

Діаграма варіантів використання представлена на рисунку 3.7.

					IT51.090БАК.002 ПЗ	Аркуш
						42
Зм.	Арк.	№ документа	Підпис	Дат		



Рисунок 3.7 – Uml-діаграма варіантів варіантів браузерної онлайн-гри на архітектурі клієнт-сервер

3.5 Діаграма послідовності для програмного забезпечення браузерної онлайн-гри на архітектурі клієнт-сервер

Наведемо діаграму послідовності дій (рисунок 3.9). Спочатку користувач розпочинає гру. Користувач створює об'єкти та виконує рухи (вправо, вліво, вперед, назад). Далі дії виконує адміністратор.

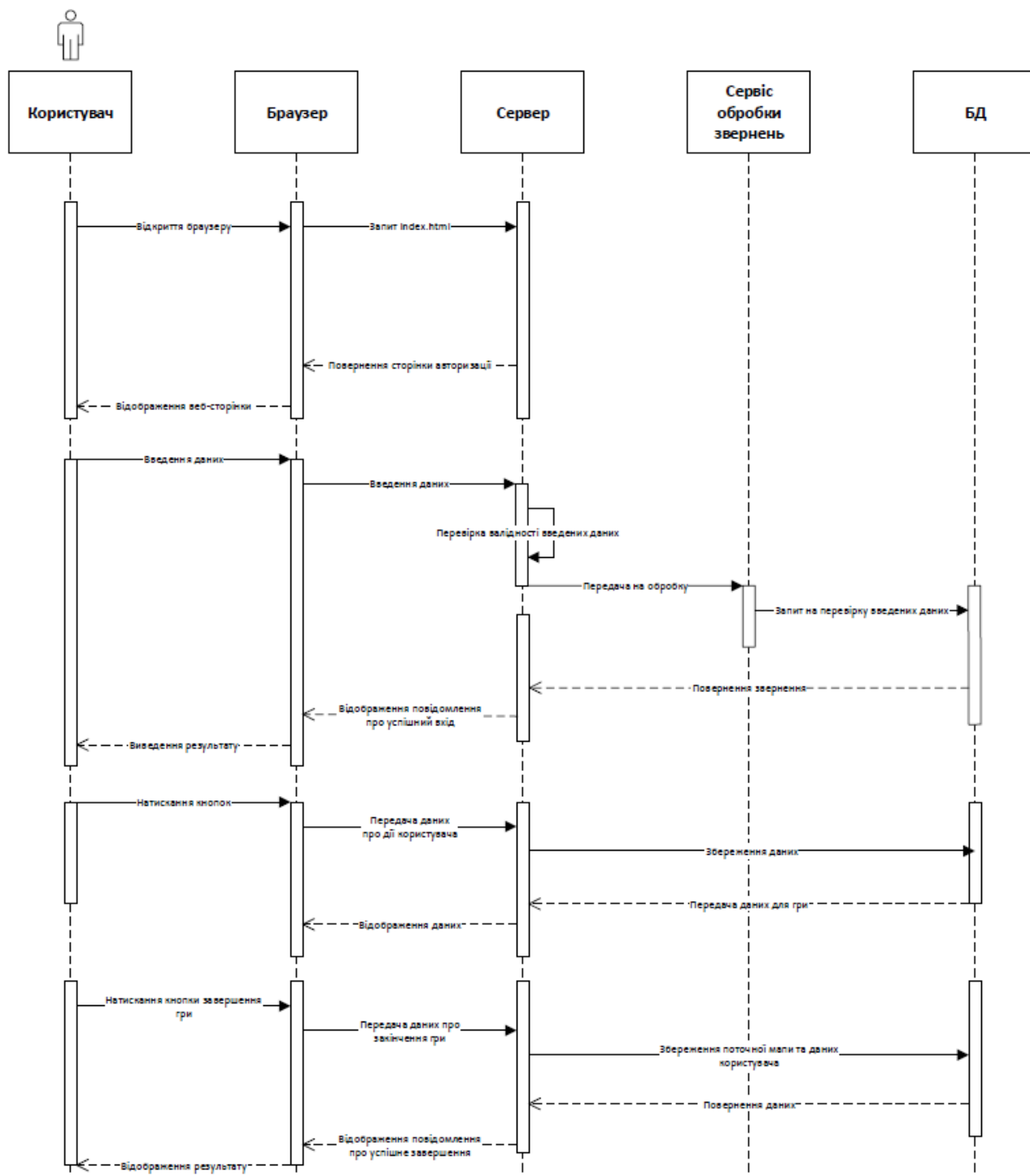


Рисунок 3.9 – Uml-діаграма послідовності дій браузерної онлайн-гри на архітектурі клієнт-сервер

3.6 Діаграма діяльності для програмного забезпечення браузерної онлайн-гри на архітектурі клієнт-сервер

Діаграма діяльності користувача та адміністратора описує стани які створюються у випадку використання системи – браузерної онлайн-гри на архітектурі клієнт-сервер (рисунок 3.10 та 3.11). У діаграмі існує два спеціальні стани: початковий (Start) – запуск системи, і кінцевий (Stop) – вихід з системи.

					IT51.090БАК.002 ПЗ	Аркуш
						45
Зм.	Арк.	№ документа	Підпис	Дат		

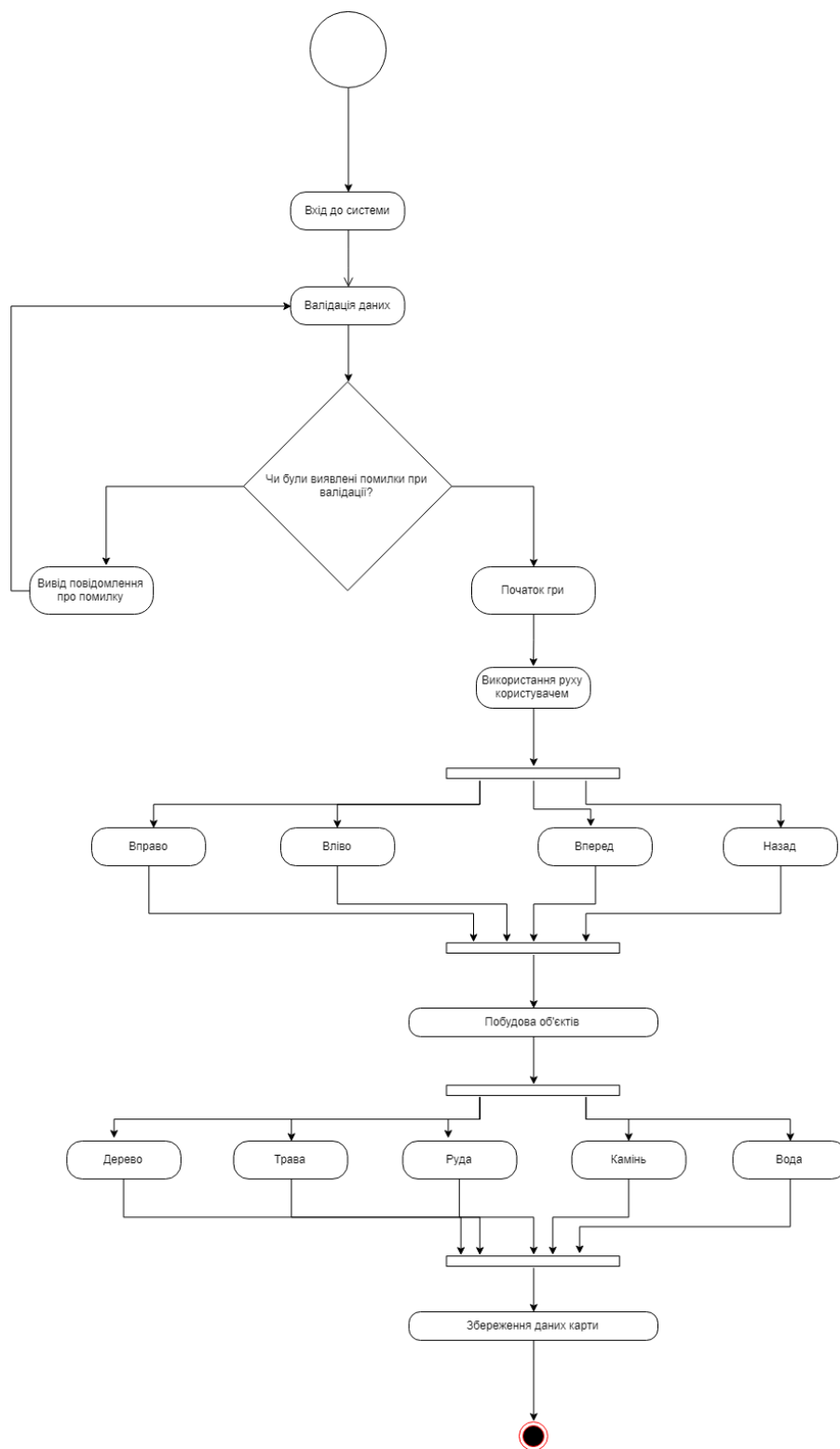


Рисунок 3.10 – Uml-діаграма діяльності користувача браузерної онлайн-гри на архітектурі клієнт-сервер

3.7 Діаграма розгортання для програмного забезпечення браузерної онлайн-гри на архітектурі клієнт-сервер

Діаграма розгортання браузерної онлайн-гри на архітектурі клієнт-сервер (рисунок 3.11), відображає обчислювальні вузли під час роботи програми, компоненти, та об'єкти, що виконуються на цих вузлах.

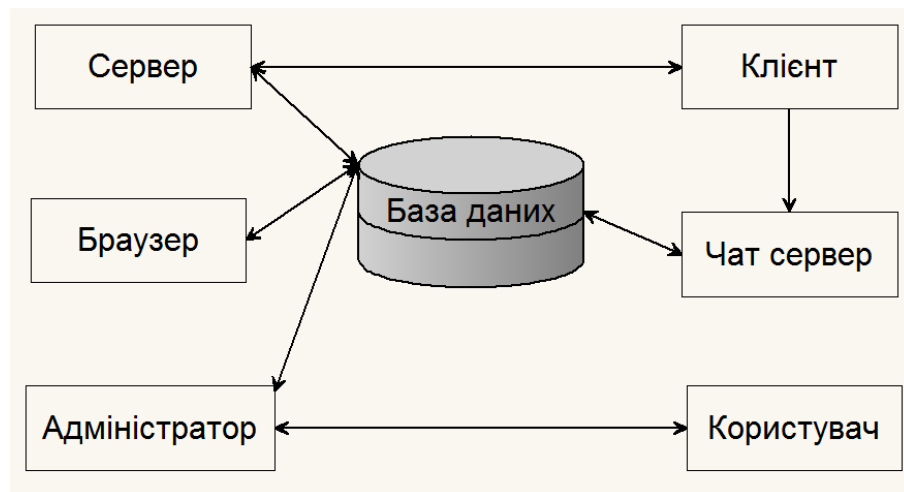


Рисунок 3.11 – Uml-діаграма розгортання користувача браузерної онлайн-гри на архітектурі клієнт-сервер

3.8 Статичний аналіз коду програмного забезпечення браузерної онлайн-гри на архітектурі клієнт-сервер

Аналіз та діагностування коду - це процес виявлення помилок і недоліків у вихідному коді програм. Статичний аналіз можна розглядати як автоматизований процес огляду коду.

У більшості випадків аналіз виконується на деякій версії вихідного коду, а в інших випадках – в деякій формі об'єктного коду.

На рисунку 3.13 та 3.14 зображено аналіз та діагностика коду, виконаний за допомогою Visual Studio 2017.

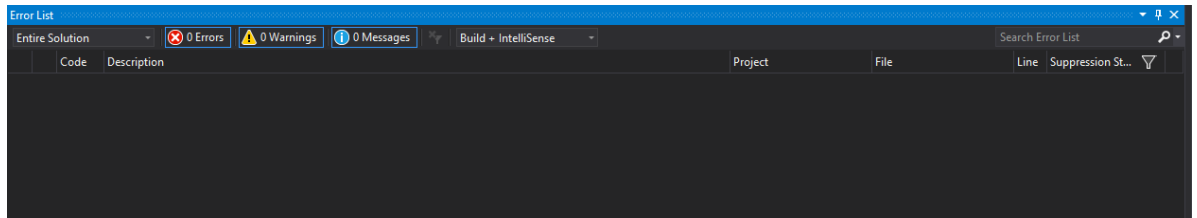


Рисунок 3.13 – Аналіз вбудованим рішенням Visual Studio 2017

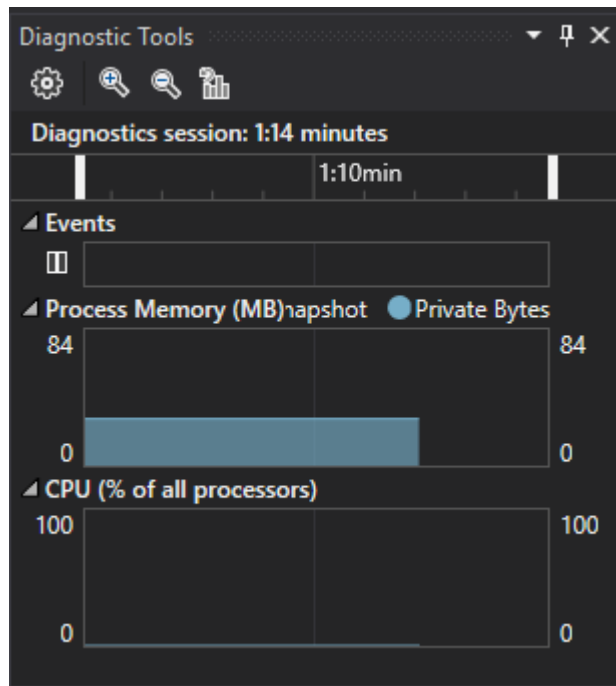


Рисунок 3.14 – Діагностика вбудованим рішенням Visual Studio 2017

На рисунку 3.15 виконане остаточне діагностування коду браузерної онлайн-гри на архітектурі клієнт-сервер вбудованим рішенням Visual Studio 2017.

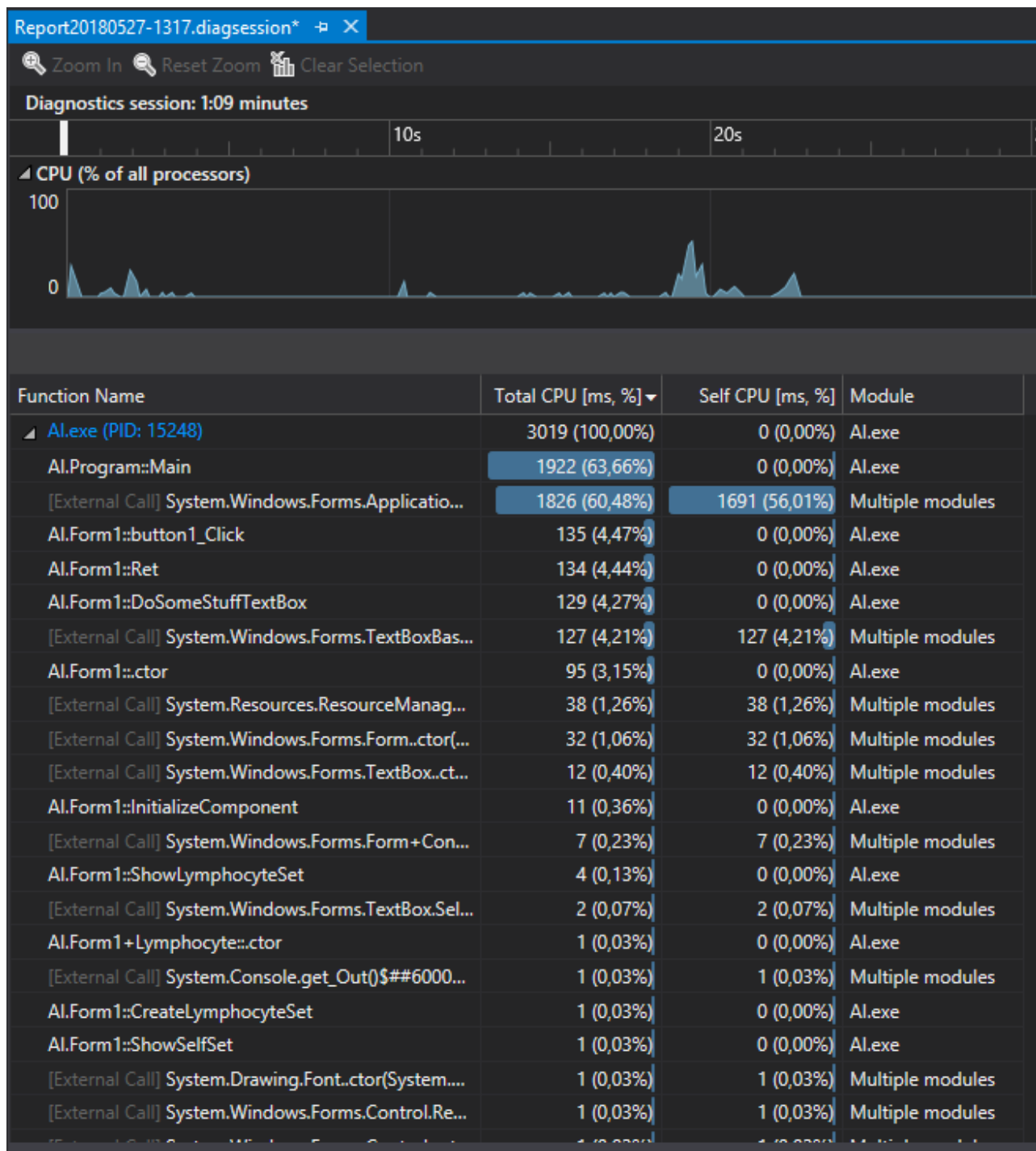


Рисунок 3.15 – Остаточне діагностування коду вбудованим рішенням Visual Studio 2017

Висновок

Розроблено програмне забезпечення браузерної онлайн-гри на архітектурі клієнт-сервер.

Застосування розробленої інформаційної технології дозволяє підвищити достовірність процесу розроблення браузерної онлайн-гри на архітектурі клієнт-сервер в порівнянні з відомими інформаційними технологіями та засобами.

Набула подальшого розвитку інформаційна технологія процесу розроблення браузерної онлайн-гри на архітектурі клієнт-сервер.

Інформаційна технологія дозволяє здійснювати якісне розроблення браузерної онлайн-гри на архітектурі клієнт-сервер .

Для програмної реалізації була використана мова програмування C#, а засобом реалізації було обрано середовище програмування Visual Studio 2017, також було використано HTML5 та Node.js.

В результаті виконаного наукового дослідження розроблено браузерну онлайн-гру на архітектурі клієнт-сервер.

					IT51.090БАК.002 ПЗ	Аркуш
						50
Зм.	Арк.	№ документа	Підпис	Дат		

4 ІНСТРУКЦІЯ КОРИСТУВАЧА БРАУЗЕРНОЇ ОНЛАЙН ГРИ НА ОСНОВІ АРХІТЕКТУРИ КЛІЄНТ-СЕРВЕР

4.1 Призначення браузерної онлайн-гри на основі архітектури клієнт-сервер

Браузерна онлайн-гра на основі архітектури клієнт-сервер є сучасним освітнім продуктом, виконана із застосуванням сучасних електронних технологій навчання, побудована на матеріалах природничо-наукового напрямку, так і матеріалів, що розкривають основи технологій, та застосовуються в наноіндустрії. Змістовна область розроблених освітніх завдань переважно відноситься до природно-науковому освітнього профілю STEM, включаючи основи нанотехнологій і технопідприємництво.

Браузерна онлайн-гра на основі архітектурі клієнт-сервер – комплекс програмного забезпечення для електронних обчислювальних машин, створений для організації ігрового процесу, зв'язку з користувачами по онлайн-грі на електронних обчислювальних пристроях з браузерів користувачів. Принциповою відмінністю освітньої комп'ютерної гри є наявність освітньої складової, інтегрованої в загальну архітектуру клієнт-сервер браузерної онлайн-гри, що містить освітні завдання та міні, інтегровані в сюжет браузерної онлайн-гри, матеріали та підказки для вирішення освітніх завдань.

4.2 Завантаження браузерної онлайн-гри на основі архітектури клієнт-сервер

Браузерна онлайн гра на основі клієнт-сервера розпочинається з входження користувачем до браузера та запуску гри вибравши відповідне завантаження карти, що зображено на рисунку 4.1.

					IT51.090БАК.002 ПЗ	Аркуш
						51
Зм.	Арк.	№ документа	Підпис	Дат		

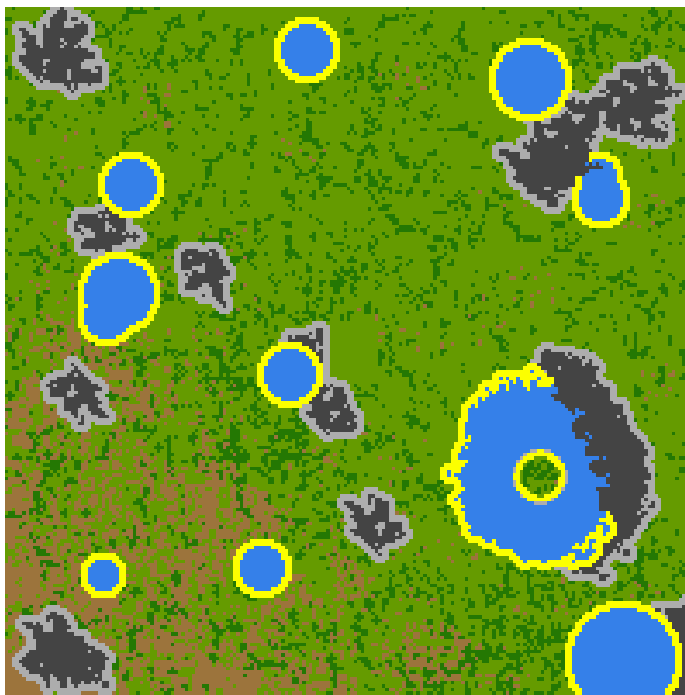


Рисунок 4.1 – Запуск браузерної онлайн гри на основі архітектури клієнт-сервер

Вибравши потрібний етап карти, відбувається відкривання та початок браузерної онлайн-гри на основі архітектури клієнт-сервер.

4.3 Початковий інтерфейс браузерної онлайн-гри на основі архітектури клієнт-сервер

Основний інтерфейс браузерної онлайн-гри на основі архітектури клієнт-сервер зображено на рисунку 4.2. Даний інтерфейс має вигляд приблизеної карти браузерної онлайн-гри та відображення ресурсів користувача.

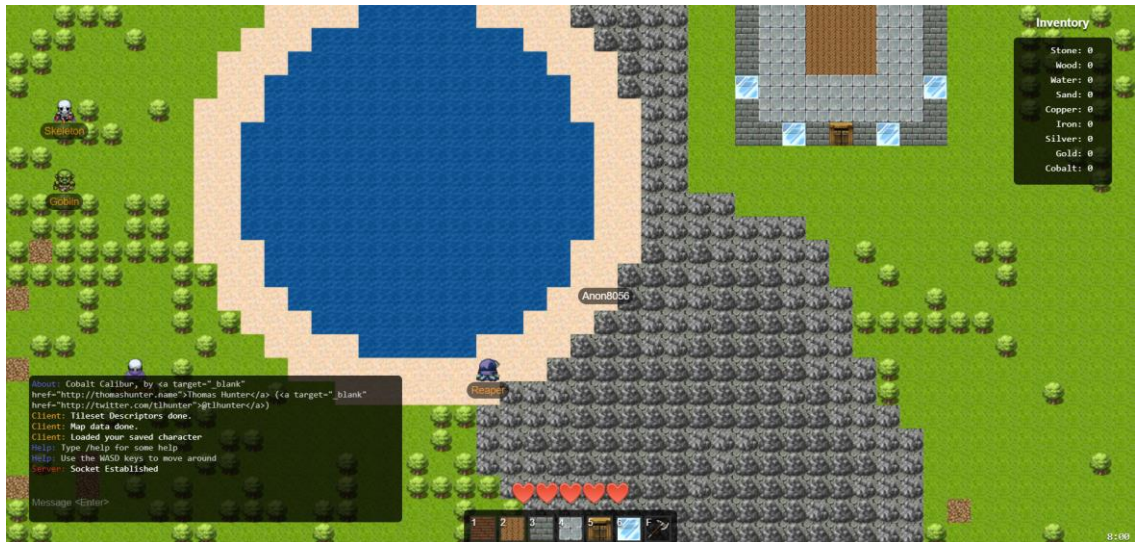


Рисунок 4.2 – Інтерфейс головної сторінки браузерної онлайн-гри на основі архітектури клієнт-сервер

4.4 Інтерфейс локації браузерної онлайн-гри на основі архітектури клієнт-сервер

У браузерній онлайн грі на основі архітектури клієнт-сервер все зроблено для зручності користувачів - приємний інтерфейс, лаконічність і легкість використання ігрових можливостей, зрозуміла навігація по локаціях. В даному програмному продукті не має яскравого дизайну, а розроблений по-справжньому якісний функціонал браузерної онлайн гри на основі архітектури клієнт-сервер.

					IT51.090БАК.002 ПЗ	Аркуш
Зм.	Арк.	№ документа	Підпис	Дат		53



Рисунок 4.3 – Інтерфейс локації браузерної онлайн-гри на основі архітектури клієнт-сервер

4.5 Інтерактивні елементи браузерної онлайн-гри на основі архітектури клієнт-сервер

Інтерактивні предмети - це елементи гри, з яким користувач браузерної онлайн гри взаємодіє (тобто, користувач може натиснути на них і отримати зворотній зв'язок в тій чи іншій формі). При наведенні курсора на такий предмет він підсвічується певним кольором.

При наведенні курсора на інтерактивний предмет, курсор може отримати додатковий маркер.

При взаємодії з інтерактивним предметом з цим маркером користувач отримає нову інформацію (відкриється інформаційне вікно або станеться додавання нового матеріалу в інфополе). Даний процес браузерної онлайн гри на основі архітектури клієнт-сервер зображено на рисунку 4.4.

					IT51.090БАК.002 ПЗ	Аркуш
Зм.	Арк.	№ документа	Підпис	Дат		54

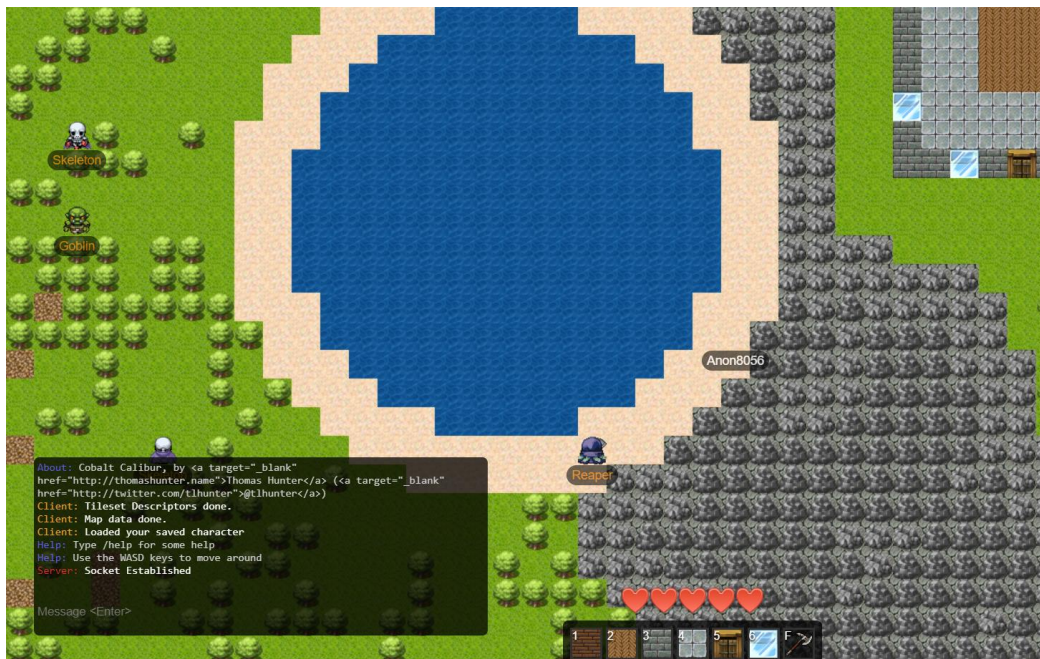


Рисунок 4.4 – Відображення інтерактивних предметів в браузерній онлайн-грі на основі архітектури клієнт-сервер

4.6 «Логи» браузерної онлайн-гри на основі архітектури клієнт-сервер

«Логи» браузерної онлайн гри на основі архітектури клієнт-сервер бувають двох видів.

Просто логи сервера, відображені на рисунку 4.5, що зберігаються в файлах. А також «логи» покупок та інших ігрових дій, що зберігаються в спеціальній базі даних / системі для гейм користувачів. Основна її ідея в тому, що вона розроблена під терабайти даних, відображена на рисунку 4.6. А також приклад «логів» в базі даних браузерної онлайн гри на основі архітектури клієнт-сервер зображено на рисунку 4.7.

```

Express      Attempting to listen on: 127.0.0.1:80
Player      Connected
Player      Disconnected
Player      Connected
Player      Connected, Name: Anon8056
Event       Corruption Spread
Player      Disconnected
Player      Connected
Player      Connected, Name: Anon8056
New Trees   1
New Grass   694
New Sand    0
Event       Time: 9:00

```

Рисунок 4.5 – «Логи» браузерної онлайн-гри на основі архітектури клієнт-сервер





Database Name ^	Storage Size	Collections	Indexes	
admin	16.4KB	0	1	
config	4.1KB	0	2	
local	16.4KB	1	1	
terraformia	143.4KB	1	1	

Рисунок 4.6 – База даних браузерної онлайн-гри на основі архітектури клієнт-сервер

local.startup_log

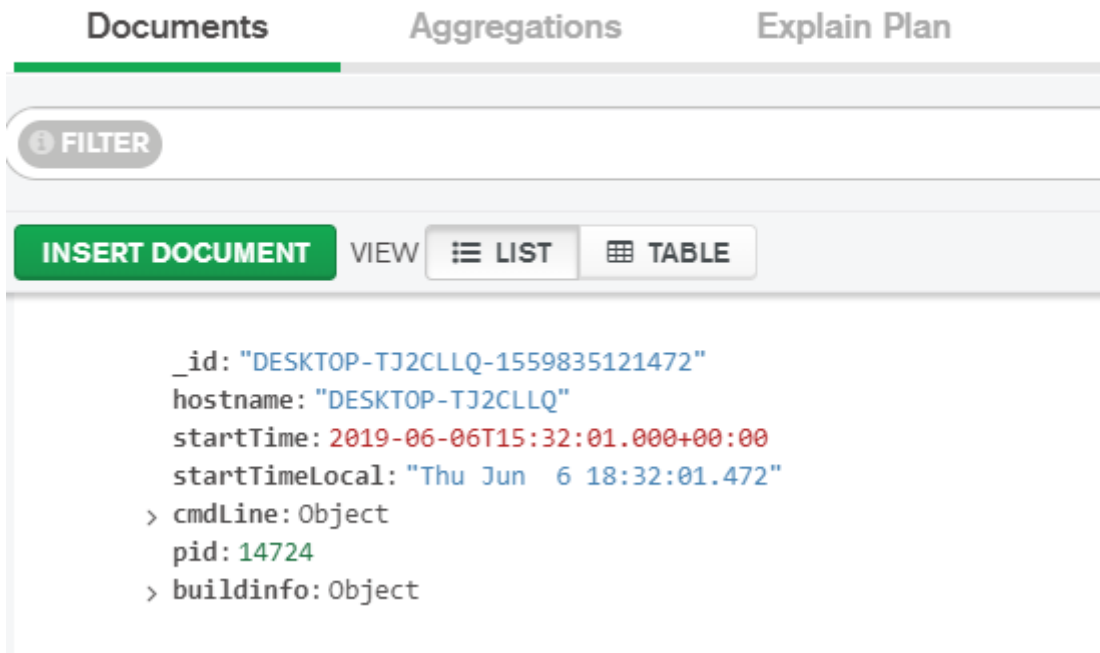


Рисунок 4.7 – «Логи» в базі даних браузерної онлайн гри

4.7 Інвентар в браузерної онлайн-гри на основі архітектури клієнт-сервер

Інвентар дозволяє користувачеві бачити, які внутрішньо ігрові предмети є в його розпорядженні. Предмети згруповані за типами по розділам . При отриманні нового предмета він автоматично потрапляє в відповідний розділ. Кожен предмет представлений іконкою, що містить в собі, рисунок 4.8:

- Назву предмета.
- Зображення предмета.
- Індикатор кількості.



Рисунок 4.8 – Інвентар браузерної онлайн гри

4.8 Чат користувачів в браузерній онлайн-грі на основі архітектури клієнт-сервер

Чат зображений на рисунку 4.9 та ділиться на кілька вкладок:

- Загальний чат: в загальному чаті видно всі повідомлення, які з'являються в даній локації, а також привати і групові повідомлення.
- Приват: повідомлення, адресовані особисто користувачу або відіслані особисто користувачами.
- Груповий чат : Показує повідомлення, адресовані групі.
- Торговий чат: для купівлі / продажу, також пошуку і пропозиції послуг.
- Системні повідомлення: повідомлення, що з'являються автоматично.
- Допомога: Тут можна задати питання і отримати роз'яснення по ігровому процесу.
- Налаштування: Дозволяє змінити налаштування чату і відображення повідомлень в зручному для користувача вигляді.

					IT51.090БАК.002 ПЗ	Аркуш
						58
Зм.	Арк.	№ документа	Підпис	Дат		


```

About: Cobalt Calibur, by Thomas Hunter (@tlhunter).
Client: Tileset Descriptors done.
Client: Map data done.
Client: Loaded your saved character
Help: Type /help for some help
Help: Use the WASD keys to move around
Server: Socket Established
Anon3063: Hello!

```

Message <Enter>

Рисунок 4.9 – Чат користувачів в браузерній онлайн-гри на основі архітектури клієнт-сервер

Висновок

Розроблено програмне забезпечення браузерної онлайн-гри на архітектурі клієнт-сервер.

Також розроблено інструкцію використання браузерної онлайн-гри на архітектурі клієнт-сервер для користувачів.

Також проаналізовану ігровий процес розробленого програмного забезпечення браузерної онлайн-гри на архітектурі клієнт-сервер.

Розглянуто призначення та завантаження браузерної онлайн гри, а також проаналізовані реалізовані інтерфесли браузерної онлайн-гри: початковий інтерфейс та інтерфейс локації.

Було визначено, що інтерактивні предмети - це елементи гри, з яким користувач браузерної онлайн гри взаємодіє, тобто, користувач може натиснути на них і отримати зворотній зв'язок в тій чи іншій формі.

Логи» браузерної онлайн гри на основі архітектури клієнт-сервер бувають двох видів.

					IT51.090БАК.002 ПЗ	Аркуш
						59
Зм.	Арк.	№ документа	Підпис	Дат		

Просто «логи» сервера, що зберігаються в файлах та «логи» покупок та інших ігрових дій, що зберігаються в спеціальній базі даних / системі для гейму користувачів.

Чат користувачів ділиться на кілька вкладок: загальний, приватний, груповий, торговий, системні повідомлення, допомога та налаштування.

					IT51.090БАК.002 ПЗ	Аркуш
						60
Зм.	Арк.	№ документа	Підпис	Дат		

ВИСНОВКИ

У роботі за результатами виконаних теоретичних та практичних досліджень розроблено браузерну онлайн гру на основі архітектури клієнт-сервер.

У першому розділі «Огляд існуючих архітектур» досліджено особливості функціонування архітектур клієнт-сервер. Аналіз містить аналіз типів архітектур клієнт-серверів. В цьому розділі були розглянуті основні класи клієнт-серверних архітектур, типи архітектур клієнт-серверних додатків, їх переваги та недоліки.

Проаналізувавши переваги та недоліки клієнт-серверних архітектур, можна зробити висновок, що для браузерної онлайн-гри є доцільним обрання тришарової архітектури, як каркасу додатка.

Другий розділ «Огляд існуючих програмних рішень та технологій» присвячений огляду та порівнянню існуючих рішень, як з точки зору інструментів розробника для створення браузерних онлайн-ігор, так і існуючих систем на ринку розробників та корпорацій. Проведено стислий огляд та аналіз технологій розроблення та впровадження браузерних онлайн-ігор. На основі нього можна зробити висновок про наявність потреби створення рішення, яке буде задовольняти потреби користувачів. Реалізація повинна ввібрати максимум найкращих рис з розглянутих технологій, а саме простий, зрозумілий інтерфейс, легкість в опануванні та потрібна кількість ресурсів застосування. У цьому розділі було описано основні технології: WebAssembly, HTML5 та Unity3D, що дозволяє якісно розроблювати браузерні онлайн-ігри різних видів

У третьому розділі «Програмна реалізація додатку» розроблено програмне забезпечення браузерної онлайн-гри на архітектурі клієнт-сервер.

Застосування розробленої інформаційної технології дозволяє підвищити достовірність процесу розроблення браузерної онлайн-гри на архітектурі клієнт-сервер в порівнянні з відомими інформаційними технологіями та засобами.

У четвертому розділі «Інструкція користувача браузерної онлайн гри на основі архітектури клієнт-сервер» розроблено програмне забезпечення браузерної

					<i>IT51.090БАК.002 ПЗ</i>	Аркуш
						61
Зм.	Арк.	№ документа	Підпис	Дат		

онлайн-гри на архітектурі клієнт-сервер. Також розроблено інструкцію використання браузерної онлайн-гри на архітектурі клієнт-сервер для користувачів. Також проаналізовану ігровий процес розробленого програмного забезпечення браузерної онлайн-гри на архітектурі клієнт-сервер.

					IT51.090БАК.002 ПЗ	Аркуш
						62
Зм.	Арк.	№ документа	Підпис	Дат		

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. M. Zuzcak, T. Sochor. Behavioral analysis of bot activity in infected systems using honeypots. In: Communications in Computer and Information Science: Springer, Cham, vol. 718, pp. 118-133, 2017.
2. A. Y. Hatata, G. Osman, M. M. Aladl, A review of the clonal selection algorithm as an optimization method. Leonardo Journal of Sciences (LJS), № 30, 2017, pp. 1-14.
3. G. Dudek. An artificial immune system for classification with local feature selection, IEEE Transactions on Evolutionary Computation, vol. 16, № 6, 2012, pp. 847-860.
4. Малюк А. Теория защиты информации / А. Малюк. – Телеком, 2012. – 184 с. – ISBN 5-93517-197-X.
5. Allix K. Large-scale machine learning based malware detection: confronting the 10-fold cross validation scheme with reality / K. Allix, T. F. Bissyandé, Q. Jérôme, J. Klein, Y. Le Traon // Proceedings of the 4th ACM conference on Data and application security and privacy, ACM, 2014, pp. 163–166.
6. Gowher M.P. Genetic algorithms in intrusion detection systems: A survey / M.P. Gowher, S. Kumar, 2014, pp. 233-240.
7. Ren Chevance. Server Architectures: Multiprocessors, Clusters, Parallel Systems, Web Servers, Storage Solutions / Ren Chevance – Elsevier/Digital Press, 2005. – 690 p
8. Erich Gamma. Design Patterns: Elements of Reusable Object-Oriented Software / Erich Gamma, Richard Helm, Ralph Johnson et al. – AddisonWesley Professional, 1994. – 458 p.
9. Martin Fowler. Patterns of Enterprise Application Architecture / Martin Fowler. – Addison-Wesley Professional, 2002. – 649 p.
10. Len Bass. Software Architecture in Practice / Len Bass, Paul Clements, Rick Kazman. – Addison-Wesley Professional, 2012. – 334 p.

					<i>IT51.090БАК.002 ПЗ</i>	Аркуш
						63
Зм.	Арк.	№ документа	Підпис	Дат		

11. Гонсалвес Э. Изучаем Java EE 7 / Э. Гонсалвес. – Appress, 2016. – 673 р.
12. Peter Herzum. Business Component Factory : A Comprehensive Overview of Component-Based Development for the Enterprise / Peter Herzum, Oliver Sims. - Wiley, 1999 – 257 p.
13. Jose Sandoval. RESTful Java Web Services / Jose Sandoval. – PACT publishing, 2009. – 328 p.
14. Layered Application, Microsoft Developers Network – Режим доступу: <https://msdn.microsoft.com/en-us/library/ff650258.aspx> - Дата тоступу: 24.05.2019.
15. A multi-tier architecture for building RESTful Web services – Режим доступу: <http://www.ibm.com/developerworks/library/wa-aj-multitier/> - Дата доступу 27.05.2019.
16. Jim Webber. REST in Practice, Hypermedia and Systems Architecture / Jim Webber, Savas Parastatidis, Ian Robinson. – O`Reilly Media, 2010 – 287 p.
17. W3C, SOAP Messages with Attachments – Режим доступу: <https://www.w3.org/TR/SOAP-attachments/> - Дата доступу: 27.05.2019.
18. Офіційна сторінка XML-RPC – Режим доступу: <http://xmlrpc.scripting.com/> - Дата доступу: 30.05.2019.
19. 3. Elearning 101 concepts, trends, applications – Режим доступу: <http://www.talentlms.com/elearning/> - Дата доступу: 4.06.2019.
20. Understanding the Three-Tier Architecture. Oracle Documentation – Режим доступу: <http://docs.oracle.com/undtldev010> – Дата доступу: 15.05.2019.
21. Офіційний сайт Spring Framework – Режим доступу: <https://spring.io/> - Дата доступу: 14.06.2019.

					<i>IT51.090БАК.002 ПЗ</i>	Аркуш
						64
Зм.	Арк.	№ документа	Підпис	Дат		